CAR-TR-781
CS-TR-3506

DAAH04-93-G-0419
July 1995

3-D Model-Based Image Stabilization Using Multiple Visual Cues

Y.S. Yao
P. Burlina
R. Chellappa
T.H. Wu

Computer Vision Laboratory
Center for Automation Research
University of Maryland
College Park, MD 20742-3275

Abstract

This paper studies the problem of image stabilization, defined here as the process of generating a compensated video sequence where image motion resulting from camera motion has been partially or totally removed. The scheme combines various visual cues such as points and horizon lines, and relies on an Extended Kalman Filter for the estimation of parameters of interest. We study both calibrated and uncalibrated stabilization cases. We address the issues of local versus global stabilization. We consider the problem of the selection of model dynamics for the estimation of warping parameters and illustrate the use of kinetic models for the selective removal of oscillatory motion. Experimental results from video sequences generated from off-road vehicle platforms show good performance of the stabilization schemes.

Keywords: Image stabilization, motion analysis, image warping, integration of visual cues, kinematic and kinetic laws

The support of the Advanced Research Projects Agency (ARPA Order No. A422) and the Army Research Office under Grant DAAH04-93-G-0419 is gratefully acknowledged, as is the help of Sandy German in preparing this paper.

1 Introduction

Image stabilization is a key preprocessing step in dynamic image analysis and deals with the removal of unwanted image motion in a video sequence. Depending on the final application, this unwanted motion may correspond to part or all of the image flow generated by the 3-D camera motion. If the ultimate goal is feedback control of the mechanical system, the undesired image motion corresponds to the high frequency oscillatory motion of the platform hosting the camera. Image stabilization then has a purpose similar to that of mechanical stabilization. If the task consists of detecting independently moving objects, the unwanted image motion is instead that motion resulting from camera rotation. If the goal is mosaicking and change detection [2, 8], the unwanted image motion corresponds to the total image flow.

Stabilization is here principally understood as the warping of video sequences for the removal of image motion due to camera rotation. This procedure is important for various reasons. The most significant reason is that the rotational flow does not convey structural information and image motion due to camera translation can often be confused with flow resulting from camera rotation [13]. Image stabilization is therefore useful for motion analysis, structure from motion [9, 10], as well as the recovery of the Focus of Expansion (FOE) and other structural information (such as time to collision). After performing stabilization, simple and effective independent motion detection mechanisms can be employed [12]. In addition, it is important for visual control, as well as for other image exploitation tasks such as registration, object detection [20], automatic target recognition, autonomous vehicle navigation [6], and model-based compression [11].

While achieving stabilization is essential for subsequent motion analysis and recognition tasks, implementing stabilization electronically[1] is also important for designing systems that are viable commercially.[2] Recently a few promising approaches to image stabilization have been proposed [6, 9, 15]. In [9], stabilization compensates for the rotational motion using a quasi-projective transformation whose parameters are computed from an assumed planar patch. This operation is performed for subsequent motion/structure recovery. Several methods using normal flow or feature trajectories based on 2-D similarity transformations or 3-D motion models are proposed in [6]. In [15], image stabilization is achieved by first aligning line segments extracted from an image

[1] Mechanical stabilization can be done using inertial sensors and a camera mounted on an isolated platform; however, this results in prohibitively costly systems.

[2] For example, some systems available in existing cameras use 2-D models to compensate for certain types of translational image motion with low precision [14].

sequence with the absolute vertical direction. Disparities between two successive frames are then compensated by 2-D linear translations. This paper studies the use of combined visual cues and dynamical models for the stabilization of calibrated or uncalibrated image sequences.

Parameters relevant for image warping are estimated by combining information from different tracked tokens, namely points and horizon lines. These parameters are simply the camera rotational velocity if intrinsic camera parameters are available, or the projectivity coefficients, in the uncalibrated case. Image plane displacements of distant feature points may unambiguously characterize rotational motion. However, such points are sometimes difficult to detect and track, due to the absence of sufficient intensity gradient information. For the same reason flow-based methods suffer from a lack of available visual information. Horizon lines, when present, constitute on the other hand very strong visual cues, requiring relatively simple operations for their tracking. These tokens must, however, be combined so as to remove all ambiguity concerning camera motion. These issues are addressed in the next section.

Image stabilization is a process closely related but not equivalent to image registration. Registration techniques can be extended for stabilization purposes. Image stabilization is inherently different, however, in that it allows the use of dynamical information over long temporal windows. In actual applications, cameras are often rigidly mounted on platforms. The rotation of the camera therefore arises from the rotational movement of the host at all times. It is possible to employ a kinetic law which captures the rotation of the platform to model the temporal behavior of the parameters of interest. The study of the resulting warping parameter dynamics therefore occupies an important place in our analysis.

Subsequently, we address the important issue of the selection of an appropriate dynamic model suitable for exploiting the temporal information in a sequence. We evaluate analytically the use of kinetic versus kinematic laws for the estimation of rotational motion components. We discuss the conditions under which the use of simpler kinematic laws yields satisfactory performance.

When dynamic laws are available, the selective removal of high frequency oscillatory motion components becomes a relevant and legitimate problem, a point which has not yet received attention in the literature. This question is carefully examined in later sections. For instance, in applications such as teleoperation, it is desirable to provide steering and climbing impressions to the teleoperator, while removing non-smooth motion. By considering the kinetic law for the host, we obtain additional insight into this problem and show conditions under which selective stabilization may be achieved.

These analytical results are applied to the stabilization of images acquired from off-road vehicles, for which the rotational motion is significant. Specifically, to stabilize a sequence, horizon lines are first extracted from each frame and distant feature points close to the horizon are detected. Both types of image primitives are tracked over the sequence. The matched lines and points thereafter form a set of visual cues. Observations are then used along with a kinematic law to estimate the needed warping parameters. Based on the estimated warping parameters, a stabilized sequence is generated. The main contributions of this paper are summarized as follows:

- *Recovery of Warping Parameters*: We study the observability of warping parameters from different image primitives (points and lines at infinity) in both calibrated and uncalibrated cases.

- *Sensitivity*: We investigate the sensitivity of the warping parameters with respect to the intrinsic parameters in uncalibrated cases.

- *Warping Schemes*: We discuss the equivalence of local and global stabilization schemes. We analyze the nature of the residual motion in the resulting warped sequence.

- *Dynamic Models*: We compare simple kinematic versus more complex kinetic laws in describing the behavior of the camera motion in the presence of sufficient visual information. The use of kinetic laws in *selective stabilization*, i.e. the removal of oscillatory motion components, is presented as well.

- *Integration of visual cues*: Different visual cues are integrated for estimating the rotational motion in a sequence. Details are given on the implementation of the Extended Kalman Filter (EKF) and on feature tracking.

The organization of this paper is as follows. Section 2 discusses warping parameter observability from different visual cues. Residual motion of warped sequences is analyzed in Section 3. Section 4 addresses selective stabilization and compares different dynamics-based estimators for the estimation of rotation. An integrated approach exploiting temporal information as well as multiple visual cues is presented in Section 5. Section 6 reports experimental results on real image sequences. Conclusions are given in Section 7.

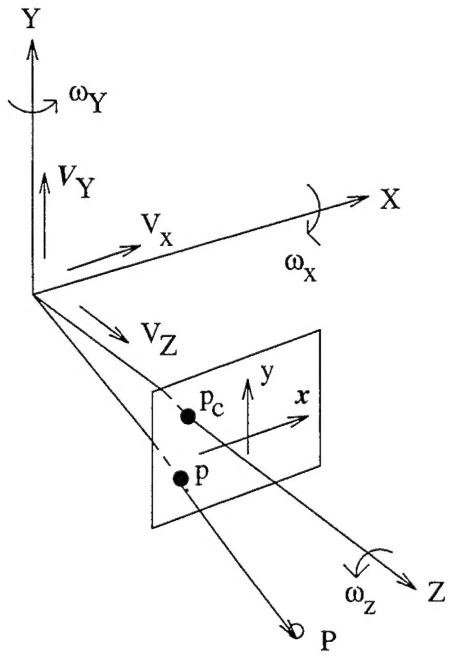


Figure 1: The camera motion and imaging model.

2 Model-guided Image Warping Schemes

This section addresses the observability of parameters used for image stabilization in both calibrated and uncalibrated cases. We address parameter recovery from points and horizon lines. Consider the scenario shown in Figure 1 where a camera undergoes rotation with instantaneous angular velocity $\boldsymbol{\omega} : (\omega_x, \omega_y, \omega_z)^T$, and translation with linear velocity $\mathbf{V} : (V_x, V_y, V_z)^T$. Let $\mathbf{P} : (X, Y, Z)^T$ denote the 3-D position of a scene point with respect to the camera, and $\mathbf{p} : (x, y)^T$ the image plane coordinates of the corresponding projection point. The relative motion of the scene point with respect to the camera is then described by

$$\dot{\mathbf{P}} = -\boldsymbol{\omega} \times \mathbf{P} - \mathbf{V} \tag{1}$$

Assuming that the perspective projection is used as an imaging model, $\mathbf{p}$ is related to $\mathbf{P}$ as follows:

$$\mathbf{p} = \mathcal{P}(\mathbf{P}) + \mathbf{p}_c \tag{2}$$

where $\mathbf{p}_c$ is the intersection of the optical axis with the image plane and $\mathcal{P}$ denotes the perspective projection operator, i.e.

$$\mathcal{P}(\mathbf{P}) = \begin{bmatrix} f_c \frac{X}{Z} \\ f_c \frac{Y}{Z} \end{bmatrix} \tag{3}$$

with f_c the focal length. Consequently, the image motion arising from the camera movement satisfies (see Appendix A for derivations)

$$\dot{\mathbf{p}} = \underbrace{[f_c^{-1}(\mathbf{p}-\mathbf{p}_c)\boldsymbol{\omega}_{xy}^{\perp T}(\mathbf{p}-\mathbf{p}_c) + \boldsymbol{\Omega}_z(\mathbf{p}-\mathbf{p}_c) + f_c\boldsymbol{\omega}_{xy}^{\perp}]}_{\dot{\mathbf{p}}_r} + \underbrace{[(\mathbf{p}-\mathbf{p}_{\text{foe}})\frac{1}{\tau}]}_{\dot{\mathbf{p}}_t} \tag{4}$$

where $\mathbf{p}_{\text{foe}} = \mathcal{P}(\mathbf{V}) + \mathbf{p}_c$ and $\tau = Z/V_z$ respectively correspond to the Focus of Expansion (FOE) and the time to collision to the imaged point; $\boldsymbol{\Omega}_z$ is the 2×2 skew-symmetric matrix related to the rotational velocity component along the optical axis

$$\boldsymbol{\Omega}_z = \begin{bmatrix} 0 & \omega_z \\ -\omega_z & 0 \end{bmatrix}; \tag{5}$$

and $\boldsymbol{\omega}_{xy}^{\perp}$ is an image vector orthogonal to the projection of the instantaneous angular velocity component parallel to the image plane,

$$\boldsymbol{\omega}_{xy}^{\perp} = \begin{bmatrix} -\omega_y \\ \omega_x \end{bmatrix} \tag{6}$$

When image motion is described instantaneously, image velocities $\dot{\mathbf{p}}$ due to 3-D rotation of the camera are expressed as second order polynomial functions of image positions and are independent of depth. For this reason, it is a well known fact that image motion resulting from rotation can be instantaneously compensated for. Care has to be taken, however, with regard to the interpretation of the resulting derotated sequence, a point addressed in the next section. On the other hand, unless relative depth is known, or all imaged points lie on the same 3-D plane, translational motion cannot be compensated for.

Consider a distant point (i.e. let $\tau \to \infty$) and denote its position by $\mathbf{P}$. As seen from (4) such points move relative to the camera as if only rotation were present. Therefore we may equivalently assume for these points that the 3-D motion simply satisfies

$$\dot{\mathbf{P}} = -\boldsymbol{\omega} \times \mathbf{P} \tag{7}$$

For off-road vehicle navigation, or images taken from a plane or a helicopter, horizon lines or partial profiles of objects lying far away constitute very strong visual cues. In Figure 2, consider an image horizon line denoted by $\mathcal{L}$; $\mathcal{L}$ is uniquely characterized by $\mathbf{W}$, the 3-D vector normal to the plane Π through $\mathcal{L}$ and the camera center. Since the image motion of horizon lines is explained exclusively

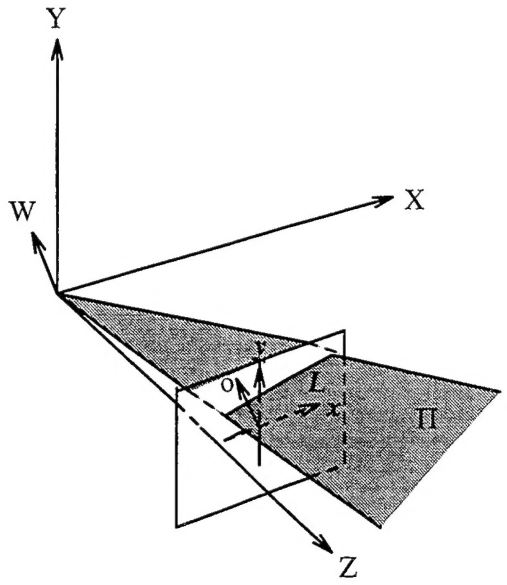


Figure 2: Geometric representation of a horizon line $\mathcal{L}$, the plane Π, the 3-D normal vector $\mathbf{W}$ and the image plane normal vector $\mathbf{o}$.

in terms of the camera rotation, it follows that the motion of the normal vector $\mathbf{W}$ is itself described by

$$\dot{\mathbf{W}} = -\boldsymbol{\omega} \times \mathbf{W} \tag{8}$$

If $\mathbf{u}$ is one solution, i.e. $\dot{\mathbf{W}} = -\mathbf{u} \times \mathbf{W}$, then $k\mathbf{W} + \mathbf{u}$ also satisfies (8) for any k. Therefore, observing the image motion of one horizon line characterizes the rotational component on plane Π only. There is indeterminacy along the direction $\mathbf{W}$. Given one observation, the set of possible solutions of (8) describes an affine line in 3-D rotational parameter space. When only one horizon line is observed, rotation may be determined in the Least Square (LS) sense which corresponds to the camera motion with least energy that explains the image motion of that particular line. In this case rotational motion inducing image motion along the line feature itself (lateral motion) is not always totally compensated for (since it is assumed to be zero), and other lines or geometrical cues near the horizon can be used to qualify lateral motion for full stabilization. Indeterminacy exists also if only one distant point is observed. In this case, this indeterminacy involves the rotational component along a ray from the image center to this point. Quantitatively, however, lines and points carry equal amounts of information in the determination of rotation. If any combination of two of these features is observed, rotation can be fully characterized, except in some degenerate cases for which $\mathbf{W} \times \mathbf{P} = \mathbf{0}$, which in practice cannot occur unless the observer possesses an unreasonably large

Field of View (FOV). Letting $\mathbf{w} = \mathcal{P}(\mathbf{W}) + \mathbf{p}_c$, one may solve the over-determined linear system

$$\boldsymbol{Q\omega} = \boldsymbol{D} \tag{9}$$

where $\mathbf{D} = [\dot{\mathbf{p}}_1^T, \ldots, \dot{\mathbf{p}}_M^T \dot{\mathbf{w}}_1^T, \ldots, \dot{\mathbf{w}}_N^T]^T$, while $\mathbf{Q}$ is a matrix derived from (4):

$$\mathbf{Q} = \begin{bmatrix} (x_1 - x_c)(y_1 - y_c) & -(f_c + (x_1 - x_c)^2) & (y_1 - y_c) \\ (f_c + f_c^{-1}(y_1 - y_c)^2) & -f_c^{-1}(x_1 - x_c)(y_1 - y_c) & -(x_1 - x_c) \\ \vdots & \vdots & \vdots \\ (x_M - x_c)(y_M - y_c) & -(f_c + (x_M - x_c)^2) & (y_M - y_c) \\ (f_c + f_c^{-1}(y_M - y_c)^2) & -f_c^{-1}(x_M - x_c)(y_M - y_c) & -(x_M - x_c) \\ (\mathrm{w}_{x_1} - x_c)(\mathrm{w}_{y_1} - y_c) & -(f_c + (\mathrm{w}_{x_1} - x_c)^2) & (\mathrm{w}_{y_1} - y_c) \\ (f_c + f_c^{-1}(\mathrm{w}_{y_1} - y_c)^2) & -f_c^{-1}(\mathrm{w}_{x_1} - x_c)(\mathrm{w}_{y_1} - y_c) & -(\mathrm{w}_{x_1} - x_c) \\ \vdots & \vdots & \vdots \\ (f_c + f_c^{-1}(\mathrm{w}_{y_N} - y_c)^2) & -f_c^{-1}(\mathrm{w}_{x_N} - x_c)(\mathrm{w}_{y_N} - y_c) & -(\mathrm{w}_{x_N} - x_c) \end{bmatrix} \tag{10}$$

Line features can therefore be combined with other tracked tokens such as distant points for stabilization.

We now discuss the image warping schemes. Irrespective of the particular trajectories of the vectors $\boldsymbol{\omega}(t)$ and $\mathbf{V}(t)$, the positions of 3-D points at two time instants can always be described by an element of the Special Euclidean group $SE(3)$ (uniquely, if the rotation center is given), i.e., there exists a total rotation $\mathbf{R} : [r_{ij}], i = 1, \ldots, 3, j = 1, \ldots, 3$ and translation $\mathbf{T} : (T_x, T_y, T_z)^T$, between any two frames, such that the 3-D point positions $\mathbf{P}_1$ and $\mathbf{P}_2$ expressed in the camera frame of reference satisfy

$$\mathbf{P}_2 = \mathbf{R}\mathbf{P}_1 + \mathbf{T} \tag{11}$$

As before, for a distant point, the contribution from the translation $\mathbf{T}$ is negligible. The image plane positions of such a point at t_1 and t_2 are then expressed as

$$\mathbf{p}_2 = (\mathbf{c}^T \mathbf{p}_1 + 1)^{-1}(\mathbf{A}\mathbf{p}_1 + \mathbf{b}) \tag{12}$$

where

$$\mathbf{A} = d^{-1} \begin{bmatrix} f_c r_{11} + x_c r_{31} & f_c r_{12} + x_c r_{32} \\ f_c r_{21} + y_c r_{31} & f_c r_{22} + y_c r_{32} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \tag{13}$$

$$\mathbf{b} = d^{-1} \begin{bmatrix} -f_c x_c r_{11} - f_c y_c r_{12} + f_c^2 r_{13} - x_c^2 r_{31} - x_c y_c r_{32} + f_c x_c r_{33} \\ -f_c x_c r_{21} - f_c y_c r_{22} + f_c^2 r_{23} - x_c y_c r_{31} - y_c^2 r_{32} + f_c y_c r_{33} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \tag{14}$$

$$\mathbf{c} = d^{-1}\begin{bmatrix} r_{31} \\ r_{32} \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \tag{15}$$

$$d = -x_c r_{31} - y_c r_{32} + f_c r_{33} \tag{16}$$

Similarly, for a distant line feature characterized by the projected normal vector $\mathbf{w}$,

$$\mathbf{w}_2 = (\mathbf{c}^T\mathbf{w}_1 + 1)^{-1}(\mathbf{A}\mathbf{w}_1 + \mathbf{b}) \tag{17}$$

As seen from (12) and (17), for distant features, their image plane motions are described by projective group operations. In fact, this is expected since points on the horizon fall into a plane at infinity and the image motion of planes is described exactly by a projectivity [1]. Consequently, if $\{r_{i,j}, i = 1,\ldots,3, j = 1,\ldots,3\}$ in $\mathbf{A}, \mathbf{b}$ and $\mathbf{c}$ are estimated from points and lines near the horizon, we can compensate for the rotation between two images using the transformation

$$\mathbf{p}_{c2} = (\mathbf{p}_2\mathbf{c}^T - \mathbf{A})^{-1}(\mathbf{b} - \mathbf{p}_2) \tag{18}$$

$$= (\tilde{\mathbf{c}}^T\mathbf{p}_2 + 1)^{-1}(\tilde{\mathbf{A}}\mathbf{p}_2 + \tilde{\mathbf{b}}) \tag{19}$$

where $\mathbf{p}_{c2}$ represents the points on the compensated image. This is also a projective transformation with parameters $\tilde{\mathbf{A}}, \tilde{\mathbf{b}}$ and $\tilde{\mathbf{c}}$, as expected by virtue of the projective group property. It is shown later that there exists only a translation between stabilized images. Therefore, (19) forms the basis of our stabilization scheme for calibrated sequences.

Note that simpler image transformations have been used in the literature for stabilization purposes: the *SE(2)* group of transformation and affine transformations, i.e. $\mathbf{p}_{c2} = \bar{\mathbf{A}}\mathbf{p}_2 + \bar{\mathbf{b}}$, are some examples used in works such as [6, 9];[3] these are essentially appropriate in cases of parallel-frontal motion.

In cases where the intrinsic parameters are unknown, we show that it is still possible to achieve image stabilization using both distant points and lines. For points, since $\mathbf{p}$ is directly measurable from the images, the projective transformation in (12) remains unchanged. However, due to the unknown $\mathbf{p}_c$ and f_c, the mapping for distant lines in (17) is no longer applicable since $\mathbf{w}$ is no longer measurable. Consider instead the measured image normal vector to a line near the horizon described in the image plane by

$$\mathbf{o}^T\mathbf{p} = 1 \tag{20}$$

[3] In [9], a second order polynomial quasi-projective transformation is assumed, $\mathbf{p}_{ct} = \mathbf{C}\mathbf{p}\mathbf{p}^T\mathbf{d} + \mathbf{A}\mathbf{p} + \mathbf{b}$ an affine transformation is then used for image warping.

It is shown in Appendix B that $\mathbf{o}$ is related to the previously defined projected normal $\mathbf{w}$ by

$$\mathbf{w} = -(1- < \mathbf{p}_c, \mathbf{o} >)^{-1} f_c^2 \mathbf{o} + \mathbf{p}_c \tag{21}$$

Using the identities $\mathbf{o}_2^T \mathbf{p}_2 = 1$ and $\mathbf{o}_1^T \mathbf{p}_1 = 1$ along with (12) we can further show that the movement of $\mathbf{o}$ also satisfies a projective transformation whose parameters are related to the inverse projective transformation $\tilde{\mathbf{A}}$, $\tilde{\mathbf{b}}$, $\tilde{\mathbf{c}}$ in (19):

$$\mathbf{o}_2 = (-\tilde{\mathbf{b}}^T \mathbf{o}_1 + 1)^{-1} (\tilde{\mathbf{A}}^T \mathbf{o}_1 - \tilde{\mathbf{c}}) \tag{22}$$

Consequently, image stabilization of uncalibrated sequences can be carried out by estimating the eight parameters in $\mathbf{A}, \mathbf{b}$ and $\mathbf{c}$ from distant points and lines. While this is possible in principle, in practice, the computation of the projectivity parameters from distant features can be unstable. To see this, consider the system obtained from (12),

$$\mathbf{G}\mathbf{f} = \mathbf{q} \tag{23}$$

where $\mathbf{f}$ is the vector including the eight projectivity parameters, $\mathbf{f} = (a_{11}, \ldots, c_2)^T$, $\mathbf{q}$ consists of the coordinates of a set of points such as $\mathbf{p}_2$, and $\mathbf{G}$ is a matrix with elements composed of the coordinates of $\mathbf{p}_1$ and $\mathbf{p}_2$. This matrix is often ill conditioned; it can be shown that the last two columns of $\mathbf{G}$ contain second order terms in the image coordinates, while the third and sixth columns contain zeroth order terms. Geometrically, since the eight parameters are computed from features lying only on a constrained region of the image (namely close to the horizon), there exist many projectivities leaving the horizon invariant, some of which are not suitable. One possible solution to this problem is to further constrain the intrinsic parameters by assuming an *approximate* value and concentrating on estimation of the rotation.

Indeed, there often are situations where the intrinsic parameters are approximately known.[4] In these cases, denote the intrinsic parameters by $\boldsymbol{\lambda} = (f_c, x_c, y_c)^T$, and let the nominal values be $\boldsymbol{\lambda}_0$. If we further assume that the eight projective coefficients vary smoothly with respect to $\boldsymbol{\lambda}$ then

$$\mathbf{f}(\boldsymbol{\lambda}) \approx \mathbf{f}(\boldsymbol{\lambda}_0) + \left.\frac{\partial \mathbf{f}}{\partial \boldsymbol{\lambda}}\right|_{\boldsymbol{\lambda}_0} (\boldsymbol{\lambda} - \boldsymbol{\lambda}_0) \tag{24}$$

$$= \mathbf{f}(\boldsymbol{\lambda}_0) + \mathbf{J}(\boldsymbol{\lambda}_0)(\boldsymbol{\lambda} - \boldsymbol{\lambda}_0) \tag{25}$$

where $\mathbf{J}$ is the Jacobian matrix. When the elements of $\mathbf{J}$ are small, the effect of the imperfect knowledge of the intrinsic parameters is negligible for stabilization purposes. A small error in

[4] In fact, the errors in these parameters can be moderate, as shown in the experiments later.

the assumed intrinsic parameters will still lead to acceptable stabilization results. For example, consider the sensitivity of b_1 with respect to f_c. Then from (14) and set $(x_c, y_c) = (0, 0)$, we have

$$\frac{\partial c_1}{\partial f_c} = r_{13} \cdot r_{33} \tag{26}$$

$$\approx -\omega_y \tag{27}$$

where (27) is typically true since the rotation between two consecutive frames is small, say on the order of 10^{-3}. Consequently, if the error in f_c is within 100 pixels, the error in b_1 will be within one pixel. Therefore, we can concentrate on estimating the three rotational parameters as in the case of calibrated sequences. It will be seen later that this assumption holds in real applications.

3 Residual Motion Analysis

We analyze here the nature of the resulting sequence when the image warping described in the previous section is applied. We further compare two types of stabilization: local stabilization and global stabilization. For convenience, denote the sequence of original images by $\mathcal{S} = \{I_0, I_1, \ldots, I_n\}$ and call the compensated sequence $\mathcal{S}_c = \{I_0, I_{c1}, \ldots, I_{cn}\}$. To obtain a translation-only sequence $\mathcal{S}_c$, either a local or global stabilization scheme can be utilized. The local approach directly compensates for the rotation between I_{k+1} and I_{ck} to generate I_{ck+1}, while the global method computes the rotation between I_{k+1} and I_k.

Consider first the residual motion in a sequence generated by a local stabilization scheme. Since every frame I_{k+1} is directly stabilized with respect to the previous compensated image I_{ck}, we have

$$\mathbf{P}_{ck+1} = \mathbf{R}^{c^{-1}}_{k+1,k} \mathbf{P}_{k+1} \tag{28}$$

where $\mathbf{R}^c_{k+1,k}$ denotes the rotation between I_{ck} and I_{k+1}, while $\mathbf{P}_{k+1}$ and $\mathbf{P}_{ck+1}$ respectively represent the 3-D coordinates of a scene point relative to the camera coordinate system in I_{k+1} and I_{ck+1}. Also, recall that the motion of $\mathbf{P}_k$ in the original sequence is described by

$$\mathbf{P}_{k+1} = \mathbf{R}_{k+1,k} \mathbf{P}_k + \mathbf{T}_{k+1,k} \tag{29}$$

with $\mathbf{R}_{k+1,k}$ and $\mathbf{T}_{k+1,k}$ the rotation and translation between the camera frames of reference in I_k and I_{k+1}. Substituting (29) into (28) and using (28) to relate $\mathbf{P}_{ck}$ and $\mathbf{P}_k$, the residual motion between I_{ck+1} and I_{ck} can be expressed as follows:

$$\mathbf{P}_{ck+1} = \mathbf{P}_{ck} + \mathbf{R}^{c^{-1}}_{k+1,k} \mathbf{T}_{k+1,k} \tag{30}$$

Furthermore, $\mathbf{R}^c_{k+1,k}$ is related to the rotation in the original sequence by

$$\begin{aligned} \mathbf{R}^c_{k+1,k} &= \mathbf{R}_{k+1,k}\mathbf{R}^c_{k,k-1} \\ &= \cdots \\ &= \mathbf{R}_{k+1,k}\mathbf{R}_{k,k-1}\cdots\mathbf{R}_{1,0} \end{aligned} \tag{31}$$

Therefore, as expected, the compensated sequence exhibits a purely translational motion. If $\mathbf{R}^c_{k+1,k}$ is known, the compensated sequence can be used for ego-motion recovery. Observe, however, that while the magnitude of the translation in the stabilized sequence is the same as that of the translation magnitude in the original sequence, the apparent translational heading (and therefore the FOE) now rotates with the original rotational motion.

This local stabilization scheme, however, is not practical for real applications. As seen from (31), $\mathbf{R}^c_{k+1,k}$ in fact accounts for the rotation between I_0 and I_{k+1}. For large k, the motion between I_0 and I_{k+1} is likely to be large. The overlap between I_{ck} and I_{k+1} may therefore be small. The common features are more difficult to find and consequently, it is not easy to compute $\mathbf{R}^c_{k+1,k}$ reliably. Global stabilization, on the other hand, focuses on the estimation of the rotation between I_k and I_{k+1}; the stabilized image is created afterwards with respect to the reference frame, say I_0, according to

$$\mathbf{P}_{ck+1} = (\mathbf{R}_{k+1,k}\mathbf{R}_{k,k-1}\cdots\mathbf{R}_{1,0})^{-1}\mathbf{P}_{k+1} \tag{32}$$

The sequence generated by the global stabilization scheme therefore exhibits the same residual motion as the sequence obtained using the local stabilization approach. However, in contrast with the local approach, the global stabilization scheme is more likely to provide reliable estimates of the parameters of interest in real applications, since the area of overlap between I_k and I_{k+1} is greater. This scheme is therefore employed in our work.

4 Exploitation of Dynamics for Full or Selective Stabilization

The next issue involves the robust estimation of the warping parameters used in the stabilization scheme presented above. As argued earlier, the inherent nature of stabilization implies that temporal information present in the sequence should be exploited for image warping purposes. A relevant problem then becomes that of selecting suitable parameter dynamics so as to exploit this temporal information. We concentrate on the dynamics for the rotation parameters. The incorporation of the proposed dynamic laws into the estimation process is presented later in Section 5.

We consider here dynamic laws appropriate for capturing the evolution of rotational motion of the platform on which the camera is mounted. Since the movement of the platform is affected by the interaction between mechanical elements and the environment, the motion is best described by a kinetic model which emulates the behavior of the mechanical system. The use of simpler kinematic models is compared to that of more complete kinetic models. These ideas are studied for the case of a vehicle performing off-road navigation.

Furthermore, due to the close relationship between kinetic models and mechanical stabilization, the possibility of employing kinetic models for the removal of oscillatory image motion, or so-called selective stabilization, now becomes viable. It will be shown that the needs of selective stabilization are different from those of total stabilization. More specifically, in order to achieve selective stabilization, features close to the camera, providing translational information, need to be considered as well.

This section therefore starts with the description of a kinetic model of a vehicle. The presentation of this model leads to the study of selective stabilization. Subsequently, using this model, the appropriateness of kinematic laws for describing the evolution of rotational parameters is evaluated.

4.1 Kinetic Models and Selective Stabilization

The movement of a vehicle over rough terrain, in general, can be decomposed into two components: the smooth motion and the oscillatory motion. The smooth motion corresponds to the behavior of the vehicle as if the terrain were smooth; it includes translation, as well as rotation due to steering and climbing. The oscillatory motion, on the other hand, refers to the residual vehicular motion; it characterizes the response of the vehicle to the roughness of the terrain.

The removal of the unwanted oscillatory motion is important for many applications, and is referred to here as *selective stabilization*.[5] For an active vision system, the separation of oscillatory motion from smooth motion is useful for achieving fixation. Another example where selective stabilization is important is teleoperation in which the vehicle needs to be remotely controlled. An image sequence unperturbed by the oscillatory motion, while still preserving smooth motion, is highly desirable: the teleoperator needs to fully evaluate the effects of climbing and steering. In addition, while an approximate selective stabilization scheme could be attempted by simple low-pass filtering of the computed total rotational components, a scheme which more closely resembles

[5]Note that we use the term *selective stabilization* in a different sense than [3].

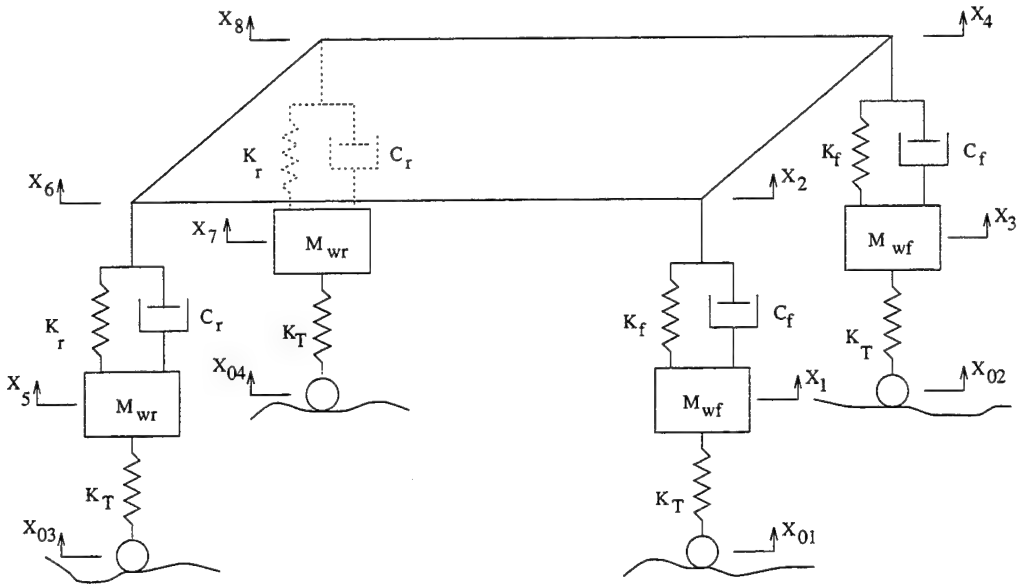


Figure 3: The four-wheel vehicle model [7].

mechanical stabilization is of more interest.

We proceed by employing a four-wheel vehicle model to account for the oscillatory vehicular movement. This model takes into account the phenomena of bounce, pitch and roll (illustrated in Figure 3), and has been widely used for the design and analysis of suspension systems. All tires are modeled by linear springs with the same stiffness coefficient K_T. M_{wf} and M_{wr} represent the masses of unsprung elements such as the front and rear wheels and their axles. K_f, C_f, K_r and C_r are the characteristics of the linear springs and shock absorbers modeling the suspension system. Assume that each tire contacts the terrain at a point at all times and the movements of unsprung elements $\{x_1, x_3, x_5, x_7\}$ are measurable (for example, by placing accelerometers on these components); then three degrees of freedom remain in this model: the bouncing displacement of the center of gravity of the sprung element x_c, the pitch angle θ and the roll angle ϕ. The high frequency yaw motion is usually small during driving; it can therefore be neglected. Note that because of the decomposition of the vehicle's movement, these oscillatory states are measured with respect to the equilibrium positions resulting from the smooth motion. Then the dynamics of the oscillatory components are expressed by [16]

$$\dot{\mathbf{x}}_{us} = \mathbf{\Phi}_{us}\mathbf{x}_{us} + \mathbf{\Gamma}_{us}\mathbf{x}_w \tag{33}$$

where $\mathbf{x}_{us}$ is the state vector consisting of the oscillatory components of interest,

$$\mathbf{x}_{us} \stackrel{\text{def}}{=} (x_c, \dot{x}_c, \theta, \dot{\theta}, \phi, \dot{\phi})^T \tag{34}$$

while $\mathbf{x}_w$ describes the measurable movements of the unsprung elements,

$$\mathbf{x}_w \stackrel{\text{def}}{=} (x_1, \dot{x}_1, x_3, \dot{x}_3, x_5, \dot{x}_5, x_7, \dot{x}_7)^T \tag{35}$$

$\mathbf{\Phi}_{us}$ and $\mathbf{\Gamma}_{us}$ respectively denote constant matrices whose entries are related to system parameters such as the mass of the sprung element, the moment of inertia, the spring characteristics, etc.

We now turn to the problem of selective stabilization using the oscillatory motion model given in (33). Assume that a camera is rigidly mounted on the vehicle, and define the stabilized coordinate system $\mathcal{V}(t)$ which describes the motion of the camera as if only smooth motion were present. Consider also the unstabilized coordinate system $\mathcal{V}'(t)$, which characterizes the total camera motion, or equivalently, the total vehicular motion. Then for a point in the scene, the movement due to the smooth motion is described by

$$\mathbf{P}_{\mathcal{V}}(t_i) = \mathbf{R}_s(t_i)[\mathbf{P}_{\mathcal{V}}(t_{i-1}) + \mathbf{T}_s(t_i)] \tag{36}$$

where $\mathbf{P}_{\mathcal{V}}(\cdot)$ is the position with respect to the stabilized coordinate system, while $\mathbf{R}_s(t_i)$ and $\mathbf{T}_s(t_i)$ account for the rotation and translation of the stabilized coordinate system between t_{i-1} and t_i. Subsequently, due to the additional oscillatory motion, the coordinates of the point with respect to the unstabilized coordinate system, $\mathbf{P}_{\mathcal{V}'}(t_i)$, are expressed as

$$\mathbf{P}_{\mathcal{V}'}(t_i) = \mathbf{R}_{us}(\theta, \phi; t_i)[\mathbf{P}_{\mathcal{V}}(t_i) - \mathbf{T}_{us}(t_i)] \tag{37}$$

where $\mathbf{R}_{us}(\theta, \phi; t_i)$ aligns both coordinate systems at the instant t_i, and $\mathbf{T}_{us}(t_i)$ represents the additional translation due to the bouncing motion. Substituting (36) into (37), we obtain the motion of the point due to the total movement of the vehicle as

$$\mathbf{P}_{\mathcal{V}'}(t_i) = \mathbf{R}_{i,i-1}\mathbf{P}_{\mathcal{V}'}(t_{i-1}) + \mathbf{R}_i(\mathbf{T}_s(t_i) + \mathbf{T}_{us}(t_{i-1})) - \mathbf{R}_{us}(\theta, \phi; t_i)\mathbf{T}_{us}(t_i) \tag{38}$$

where

$$\mathbf{R}_{i,i-1} = \mathbf{R}_{us}(\theta, \phi; t_i)\mathbf{R}_s(t_i)\mathbf{R}_{us}^{-1}(\theta, \phi; t_{i-1}) \tag{39}$$

$$\mathbf{R}_i = \mathbf{R}_{us}(\theta, \phi; t_i)\mathbf{R}_s(t_i) \tag{40}$$

The rotation $\mathbf{R}_{i,i-1}$ aligns the coordinate system $\mathcal{V}'(t_{i-1})$ with $\mathcal{V}'(t_i)$ while $\mathbf{R}_i$ aligns $\mathcal{V}(t_{i-1})$ with $\mathcal{V}'(t_i)$.

Assume that the movement of the vehicle is known up to time instant t_{i-1}, i.e. $\mathbf{R}_{us}(\theta, \phi; t_{i-1})$, $\mathbf{T}_{us}(t_{i-1})$, and $\mathbf{P}_{\mathcal{V}'}(t_{i-1})$ are known. Then, in order to differentiate between oscillatory motion

and smooth movement components, the translational components $\mathbf{T}_{us}(t_i)$ and $\mathbf{T}_s(t_i)$ need to be characterized. This observation points to a basic difference between selective and full stabilization. Different strategies therefore need to be employed for selective stabilization. More specifically, in this case, points close to the vehicle, whose image motions are affected by the translational platform movement, need to be considered. When only the motions of distant features are considered, (38) simplifies to

$$\mathbf{P}_{\mathcal{V}'}(t_i) = \mathbf{R}_{i,i-1}\mathbf{P}_{\mathcal{V}'}(t_{i-1}) \tag{41}$$

showing instead that the total rotation between two instants is observable from distant features alone.

4.2 Rotation Dynamics

We turn now to the evaluation of kinetic models (as compared with simple kinematic models) to capture the evolution of rotational motion. The simple models are sometimes casually used without further examination. We will show analytically that kinetic models are superior to kinematic models, but that kinematic models yield acceptable performance if sufficient numbers of visual observations are available.

As mentioned previously, the attitude change between two time instants is due to smooth rotation and oscillatory motion. Rotational motion components such as rotation due to climbing occur only during limited time intervals; they are therefore negligible most of the time. We will study the performance of a kinematic-law based estimator for the estimation of pitch and roll motion only. The estimator can easily be generalized to take into account all the rotational components (including the smooth motion components).

Since only oscillatory rotation is being investigated here, for convenience we can rewrite $\mathbf{\Phi}_{us}$ in (33) as

$$\mathbf{\Phi}_{us} = \begin{bmatrix} \mathbf{\Phi}_c & \mathbf{\Phi}_{cr} \\ \mathbf{\Phi}_{rc} & \mathbf{\Phi}_r \end{bmatrix} \tag{42}$$

where $\mathbf{\Phi}_c, \mathbf{\Phi}_{cr}, \mathbf{\Phi}_{rc}$ and $\mathbf{\Phi}_r$ denote constant matrices with respective sizes $2 \times 2, 2 \times 4, 4 \times 2$ and 4×4. Define a constant matrix $\mathbf{G}$ such that

$$\mathbf{y} = \mathbf{G}\mathbf{x}_{us} \tag{43}$$

with $\mathbf{y} = [\theta, \dot{\theta}, \phi, \dot{\phi}]^T$. For simplicity, the measurements $\mathbf{z}$ available for the estimation of $\mathbf{y}$ are

related to $\mathbf{y}$ by a linear transformation $\mathbf{C}$:

$$\mathbf{z} = \mathbf{C}\mathbf{y} \tag{44}$$

Assuming that the oscillatory rotational motion component is exactly described by the dynamics of $\mathbf{y}$, we compare the performance of estimators based on different dynamics when additional measurements are available. The kinetic-law based estimator captures the oscillatory motion using coefficients related to the true dynamics [5]:

$$\dot{\hat{\mathbf{y}}}_1 = \mathbf{\Phi}_r \hat{\mathbf{y}}_1 + \mathbf{K}_1[\mathbf{z} - \mathbf{C}\hat{\mathbf{y}}_1] \tag{45}$$

where $\hat{\mathbf{y}}_1$ represents the resulting estimate of $\mathbf{y}$, and $\mathbf{K}_1$ is the desired gain matrix. The kinematic-law based estimator, on the other hand, assumes no knowledge of the underlying mechanical system and therefore only employs the smooth variation dynamics

$$\dot{\hat{\mathbf{y}}}_2 = \mathbf{\Phi}\hat{\mathbf{y}}_2 + \mathbf{K}_2[\mathbf{z} - \mathbf{C}\hat{\mathbf{y}}_2] \tag{46}$$

where $\hat{\mathbf{y}}_2$ and $\mathbf{K}_2$ respectively have meanings similar to those of $\hat{\mathbf{y}}_1$ and $\mathbf{K}_1$, while

$$\mathbf{\Phi} = \begin{bmatrix} \mathbf{\Phi}_{01} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Phi}_{01} \end{bmatrix} \tag{47}$$

with $\mathbf{0}$ the 2×2 zero matrix, and

$$\mathbf{\Phi}_{01} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \tag{48}$$

Let us define the corresponding estimation errors as

$$\tilde{\mathbf{y}}_1 = \mathbf{y} - \hat{\mathbf{y}}_1 \tag{49}$$

$$\tilde{\mathbf{y}}_2 = \mathbf{y} - \hat{\mathbf{y}}_2 \tag{50}$$

It can be shown that $\tilde{\mathbf{y}}_1$ and $\tilde{\mathbf{y}}_2$ satisfy

$$\dot{\tilde{\mathbf{y}}}_1 = (\mathbf{\Phi}_r - \mathbf{K}_1\mathbf{C})\tilde{\mathbf{y}}_1 + \mathbf{u} \tag{51}$$

$$\dot{\tilde{\mathbf{y}}}_2 = (\mathbf{\Phi} - \mathbf{K}_2\mathbf{C})\tilde{\mathbf{y}}_2 + \mathbf{u} - \Delta\mathbf{\Phi}_r\hat{\mathbf{y}}_2 \tag{52}$$

where $\mathbf{u}$ is related to $\mathbf{x}_w, x_c$ and $\dot{x}_c$ by

$$\mathbf{u} = \mathbf{G}\mathbf{\Gamma}_{us}\mathbf{x}_w + \mathbf{\Phi}_{rc}\mathbf{x}_b \tag{53}$$

with $\mathbf{x}_b = [x_c, \dot{x}_c]^T$, while $\Delta\mathbf{\Phi}_r$ is due to mismatches between the assumed and true dynamics:

$$\Delta\mathbf{\Phi}_r = \mathbf{\Phi} - \mathbf{\Phi}_r \tag{54}$$

As seen from (51), if $\mathbf{K}_1$ is chosen so that the eigenvalues of $\mathbf{\Phi}_r - \mathbf{K}_1\mathbf{C}$ have negative real parts, then $\tilde{\mathbf{y}}_1$ remains bounded as long as $\mathbf{u}$ is bounded. Similarly, in addition to $\mathbf{u}$ being bounded, if $\Delta\mathbf{\Phi}\hat{\mathbf{y}}_2$ is also bounded, we can carefully choose $\mathbf{K}_2$ so that $\tilde{\mathbf{y}}_2$ is bounded. Nonetheless, $\hat{\mathbf{y}}_2$ will exhibit a larger error than $\hat{\mathbf{y}}_1$.

In sum, kinetic-law based estimators $\hat{\mathbf{y}}_1$ outperform kinematic-law based estimators $\hat{\mathbf{y}}_2$. The kinetic-law estimator, however, requires knowledge of the mechanical system parameters. These parameters are not always easily measurable. When the system parameters are unknown, the availability of sufficient numbers of visual measurements should allow for the use of the simpler kinematic law while still yielding good warping parameter estimates. The next section employs the kinematic laws for the estimation of total rotation.

5 Parameter Estimation

Various types of estimators can be employed to obtain the parameters used in our stabilization scheme. Recursive-type estimators update the estimates of the parameters whenever new information becomes available while batch-type estimators compute the estimates by processing all the information. Because of their ability to process data sequentially and their lower computational complexity, recursive-type estimators are preferred over batch estimators for real time processing. Kalman filters make use of dynamics for their estimation. We therefore provide Extended Kalman Filter (EKF) formulations for both the calibrated and uncalibrated stabilization cases.

For both calibrated and uncalibrated sequences, the algorithm only needs to estimate three rotational parameters, and the state vector $\mathbf{x}$ is simply equal to

$$\mathbf{x} = \boldsymbol{\omega} \tag{55}$$

Based on the simple kinematic law justified in the previous section, we have

$$\dot{\mathbf{x}} = 0 \tag{56}$$

Subsequently,

$$\mathbf{x}(t_{i+1}) = \mathbf{x}(t_i) \tag{57}$$

This constitutes the plant equation in our recursive estimation algorithm. We turn next to the observation equations. Assume that the tracked tokens are composed of M points (whose projection points are $\mathbf{p}_1, \ldots, \mathbf{p}_M$) and N horizon lines (each one with associated normal vector $\mathbf{w}_i, i = 1, \ldots, N$ for calibrated sequences, or alternatively with given $\mathbf{o}_i, i = 1, \ldots, N$ for uncalibrated sequences, with $\mathbf{o}_i$ defined as in (20)). The measurement vector in the calibrated case is defined by

$$\mathbf{z} = (\mathbf{p}_1^T, \ldots, \mathbf{p}_M^T, \mathbf{w}_1^T, \ldots, \mathbf{w}_N^T)^T \tag{58}$$

From (12) and (17) we can write the measurement equations as

$$\mathbf{z}(t_{i+1}) = \mathbf{h}_{i+1,i}[\mathbf{x}(t_{i+1})] + \mathbf{n}(t_{i+1}) \tag{59}$$

where $\mathbf{h}$ is a nonlinear function while $\mathbf{n}$ denotes the measurement noise. More specifically, in the calibrated case, $\mathbf{h}_{i+1,i}[\mathbf{x}(t_{i+1})]$ is expressed as

$$\mathbf{h}_{i+1,i}[\mathbf{x}(t_{i+1})] = \begin{bmatrix} (\mathbf{c}^T\mathbf{p}_1 + 1)^{-1}(\mathbf{A}\mathbf{p}_1 + \mathbf{b}) \\ \vdots \\ (\mathbf{c}^T\mathbf{p}_M + 1)^{-1}(\mathbf{A}\mathbf{p}_M + \mathbf{b}) \\ (\mathbf{c}^T\mathbf{w}_1 + 1)^{-1}(\mathbf{A}\mathbf{w}_1 + \mathbf{b}) \\ \vdots \\ (\mathbf{c}^T\mathbf{w}_N + 1)^{-1}(\mathbf{A}\mathbf{w}_N + \mathbf{b}) \end{bmatrix} \tag{60}$$

with $\mathbf{A}$, $\mathbf{b}$, and $\mathbf{c}$ expressed with respect to the state vector components as

$$\mathbf{A} = d^{-1} \begin{bmatrix} f_c r_{11} + x_c r_{31} & f_c r_{12} + x_c r_{32} \\ f_c r_{21} + y_c r_{31} & f_c r_{22} + y_c r_{32} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \tag{61}$$

$$\mathbf{b} = d^{-1} \begin{bmatrix} -f_c x_c r_{11} - f_c y_c r_{12} + f_c^2 r_{13} - x_c^2 r_{31} - x_c y_c r_{32} + f_c x_c r_{33} \\ -f_c x_c r_{21} - f_c y_c r_{22} + f_c^2 r_{23} - x_c y_c r_{31} - y_c^2 r_{32} + f_c y_c r_{33} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \tag{62}$$

$$\mathbf{c} = d^{-1} \begin{bmatrix} r_{31} \\ r_{32} \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \tag{63}$$

$$d = -x_c r_{31} - y_c r_{32} + f_c r_{33} \tag{64}$$

The measurement vector in the uncalibrated case is instead defined by

$$\mathbf{z} = (\mathbf{p}_1^T, \ldots, \mathbf{p}_M^T, \mathbf{o}_1^T, \ldots, \mathbf{o}_N^T)^T \tag{65}$$

and the observation equation then uses

$$\mathbf{h}_{i+1,i}[\mathbf{x}(t_{i+1})] = \begin{bmatrix} (\mathbf{c}^T\mathbf{p}_1 + 1)^{-1}(\mathbf{A}\mathbf{p}_1 + \mathbf{b}) \\ \vdots \\ (\mathbf{c}^T\mathbf{p}_M + 1)^{-1}(\mathbf{A}\mathbf{p}_M + \mathbf{b}) \\ (-\tilde{\mathbf{b}}^T\mathbf{o}_1 + 1)^{-1}(\tilde{\mathbf{A}}^T\mathbf{o}_1 - \tilde{\mathbf{c}}) \\ \vdots \\ (-\tilde{\mathbf{b}}^T\mathbf{o}_N + 1)^{-1}(\tilde{\mathbf{A}}^T\mathbf{o}_N - \tilde{\mathbf{c}}) \end{bmatrix} \tag{66}$$

with $\tilde{\mathbf{A}}$, $\tilde{\mathbf{b}}$, and $\tilde{\mathbf{c}}$ derived from the identities

$$\tilde{\mathbf{A}}^T = \tilde{d}^{-1} \begin{bmatrix} a_{22} - b_2c_2 & b_2c_1 - a_{21} \\ b_1c_2 - a_{12} & a_{11} - b_1c_1 \end{bmatrix} \tag{67}$$

$$-\tilde{\mathbf{c}} = \tilde{d}^{-1} \begin{bmatrix} a_{22}c_1 - a_{21}c_2 \\ a_{11}c_2 - a_{12}c_1 \end{bmatrix} \tag{68}$$

$$-\tilde{\mathbf{b}} = \tilde{d}^{-1} \begin{bmatrix} a_{22}b_1 - a_{12}b_2 \\ a_{11}b_2 - a_{21}b_1 \end{bmatrix} \tag{69}$$

$$\tilde{d} = a_{11}a_{22} - a_{12}a_{21} \tag{70}$$

With the plant and measurement equations given in (57) and (59), when horizon lines and points are tracked, the EKF scheme can be applied to recursively estimate the two-frame angular velocity. This consists of the following steps:

- **Step 1**: State and covariance propagation

$$\begin{aligned} \hat{\mathbf{x}}(t_{i+1}^-) &= \hat{\mathbf{x}}(t_i^+) \\ \mathbf{\Sigma}(t_{i+1}^-) &= \mathbf{\Sigma}(t_i^+) + \mathbf{\Sigma}_w(t_{i+1}) \end{aligned} \tag{71}$$

  where $\hat{\mathbf{x}}(t_i^+)$ and $\mathbf{\Sigma}(t_i^+)$ denote the estimates of $\mathbf{x}(t_i)$ and the associated covariances: they are obtained based on information contained in the sequence up to the i^{th} frame. $\hat{\mathbf{x}}(t_{i+1}^-)$ and $\mathbf{\Sigma}(t_{i+1}^-)$, on the other hand, are the predicted estimates of $\mathbf{x}(t_{i+1})$ and the predicted covariances respectively before the incorporation of the $(i+1)^{\text{th}}$ frame, while $\mathbf{\Sigma}_w(t_{i+1})$ is the covariance of the plant noise $\mathbf{w}(t_{i+1})$.

- **Step 2**: State and covariance update

$$\begin{aligned} \mathbf{K}(t_{i+1}) &= \Sigma(t_{i+1}^-)\mathbf{H}_{i+1,i}^T[\mathbf{H}_{i+1,i}\Sigma(t_{i+1}^-)\mathbf{H}_{i+1,i}^T + \Sigma_n(t_{i+1})]^{-1} \\ \hat{\mathbf{x}}(t_{i+1}^+) &= \hat{\mathbf{x}}(t_{i+1}^-) + \mathbf{K}(t_{i+1})\left\{\mathbf{z}(t_{i+1}) - \mathbf{h}_{i+1,i}[\hat{\mathbf{x}}(t_{i+1}^-)]\right\} \\ \Sigma(t_{i+1}^+) &= [\mathbf{I} - \mathbf{K}(t_{i+1})\mathbf{H}_{i+1,i}]\Sigma(t_{i+1}^-) \end{aligned} \tag{72}$$

where $\hat{\mathbf{x}}(t_{i+1}^+)$ is the desired estimate of $\mathbf{x}(t_{i+1})$, and $\Sigma(t_{i+1}^+)$ is the associated covariance. $\mathbf{K}(t_{i+1}), \Sigma_n(t_{i+1})$ and $\mathbf{I}$ respectively represent the gain matrix, the covariance of $\mathbf{n}(t_{i+1})$ and the identity matrix. $\mathbf{H}_{i+1,i}$, on the other hand, is the linearized approximation of $\mathbf{h}_{i+1,i}$,

$$\mathbf{H}_{i+1,i} = \left.\frac{\partial \mathbf{h}_{i+1,i}}{\partial \mathbf{x}(\mathbf{t_{i+1}})}\right|_{\hat{\mathbf{x}}(t_{i+1}^-)} \tag{73}$$

The algorithm is initialized by a batch process using an LS estimate of the rotational parameters as in

$$\min_{\mathbf{x}(t_1)} \sum_i ||\mathbf{z}(t_{i+1}) - \mathbf{h}_{i+1,i}[\mathbf{x}(t_{i+1})]||^2 \tag{74}$$

The minimization can be solved using techniques based on gradient descent, such as conjugate descent.

This concludes our discussion of parameter estimation. The next section addresses image primitive tracking. Synthetic and real experiments are reported as well.

6 Experimental Results

We first briefly describe approaches to the detection and tracking of horizon lines and points. Experimental results on two real image sequences are then presented.

6.1 Detection/Tracking of Image Primitives

The first set of visual cues for characterizing the rotation consists of horizon lines. There have been numerous approaches to tracking a set of line segments over a sequence [4, 18]. For simplicity, we only focus on tracking one line in our work, although using a set of lines is desirable in some situations.

Assuming that the lines near the horizon appear in the form of large vertical brightness derivatives, the detection of the line of interest is achieved in three steps:

- **Step 1**: Based on the histogram of the image, a binary image is generated by thresholding the original image.

- **Step 2**: In the binary image, the pixels with large vertical brightness derivatives most likely constitute points on the horizon line. A simple template matching technique is applied to identify pixels with large brightness derivatives.

- **Step 3**: A median fit method is used to robustly group identified pixels into lines. The longest line is taken to be the line of interest.

In the experiments, the above procedures are applied to each image in the sequence. The line extracted from each image is assumed to be near the horizon, and these lines are matched to each other.

The second class of inputs to the full stabilization scheme are the image plane trajectories of a set of points near the horizon. A localized tracking algorithm reported in [17] is employed to obtain feature point trajectories. In particular, a Gabor-wavelet based feature extraction algorithm is initially applied to the first image. Then, according to the detection of lines near the horizon, feature points of interest are selected to initiate the tracking algorithm.

For each feature point, the algorithm employs a trajectory model to exploit the temporal information and decomposes the sequence-tracking problem into successive two-frame tracking problems. Three steps are involved in finding the matching point of a tracked feature point in the following frame:

- **Step 1**: Based on the trajectory model, the first step uses a probabilistic data association technique to estimate the inter–frame motion. The resulting estimates are used in the next step to facilitate the process of finding the corresponding point in the following frame.

- **Step 2:** The second step applies a correlation–type matching algorithm to identify the corresponding point. The inter–frame motion estimates are first used to compensate for the rotation as well as to reduce the search area for the matching point. Various criteria [19] are then applied to find the matching point to sub–pixel accuracy.

- **Step 3:** The third step processes the temporal information contained in the new frame by updating the parameters in the trajectory model using a Kalman filter. Afterwards, the algorithm goes back to the first step and continues to track the point in the following frame.

Under the assumption that the tracked points exist at all times, the feature point tracking algorithm can identify the corresponding points over a sequence. However, in our formulation, it

is not required that the same set of points be used. Therefore, we only need to focus on finding matching points between two frames. A different set of points can be used whenever this is desirable.

6.2 Synthetic Experiments

This section tests the results of Section 4; we illustrate the performance of the kinematic law using a one-dimensional example, and show that both laws yield comparable performance when sufficient numbers of visual observations are present. Consider the scenario in which the vehicle travels along a straight path most of the time except when encountering impulse-like disturbances (bumps). Each bump is modeled by a half sine wave with a given height and width. (For simplicity, no smooth rotational motion during driving is assumed.) The excitations to the vehicle are then given by

$$x_{01}(t) = \begin{cases} b_h \sin[\frac{\pi}{b_w}(vt - z_0)] & \frac{z_0}{v} \leq t \leq \frac{z_0 + b_w}{v} \\ 0 & \text{elsewhere} \end{cases}$$

with

$$x_{02}(t) = x_{01}(t), \quad x_{03}(t) = x_{01}(t - \frac{L}{v}), \quad x_{04}(t) = x_{02}(t - \frac{L}{v})$$

where z_0, L, and v respectively denote the location of the bump, the vehicle's length and its forward speed. (Only one bump is used in this example, with $Z_0 = 1.345$ m, $b_h = 0.1$ m, $b_w = 0.2$ m, $L = 2.7$ m and $v = 13.45$ m $\cdot$ s^{-1}). The nominal values of the vehicle parameters are listed in Table 1, and the oscillatory motion is synthesized according to (33) with only the pitching motion shown in Figures 4a and 4b. During driving, a sequence of images is acquired at 20 Hz from an on-board camera (each image has size 2.0×2.0 and resolution 2000×2000), and a set of distant points is tracked; the coordinates of these points with respect to the initial camera frame of reference are listed in Table 2.

Table 1: Model parameters. M_B: the mass of the sprung element, I_{yy}: the moment of inertia along the pitch axis, W_A and W_B: the distance of the center of gravity to the front and rear ends.

M_B	I_{yy}	M_{wf}	M_{wr}	K_T
1710.0kg	1031.3kg·m^2	57.5kg	75.0kg	200.0kN·m^{-1}
K_f	K_r	C_f (C_r)	W_A	W_B
18.0kN·m^{-1}	10.0kN·m^{-1}	1.0kN·m^{-1}·s^{-1}	1.4m	1.3m

Subsequently, to compare the kinetic and and kinematic law based estimators, we design a recursive-type estimator similar to the one used in the real application to estimate the pitching

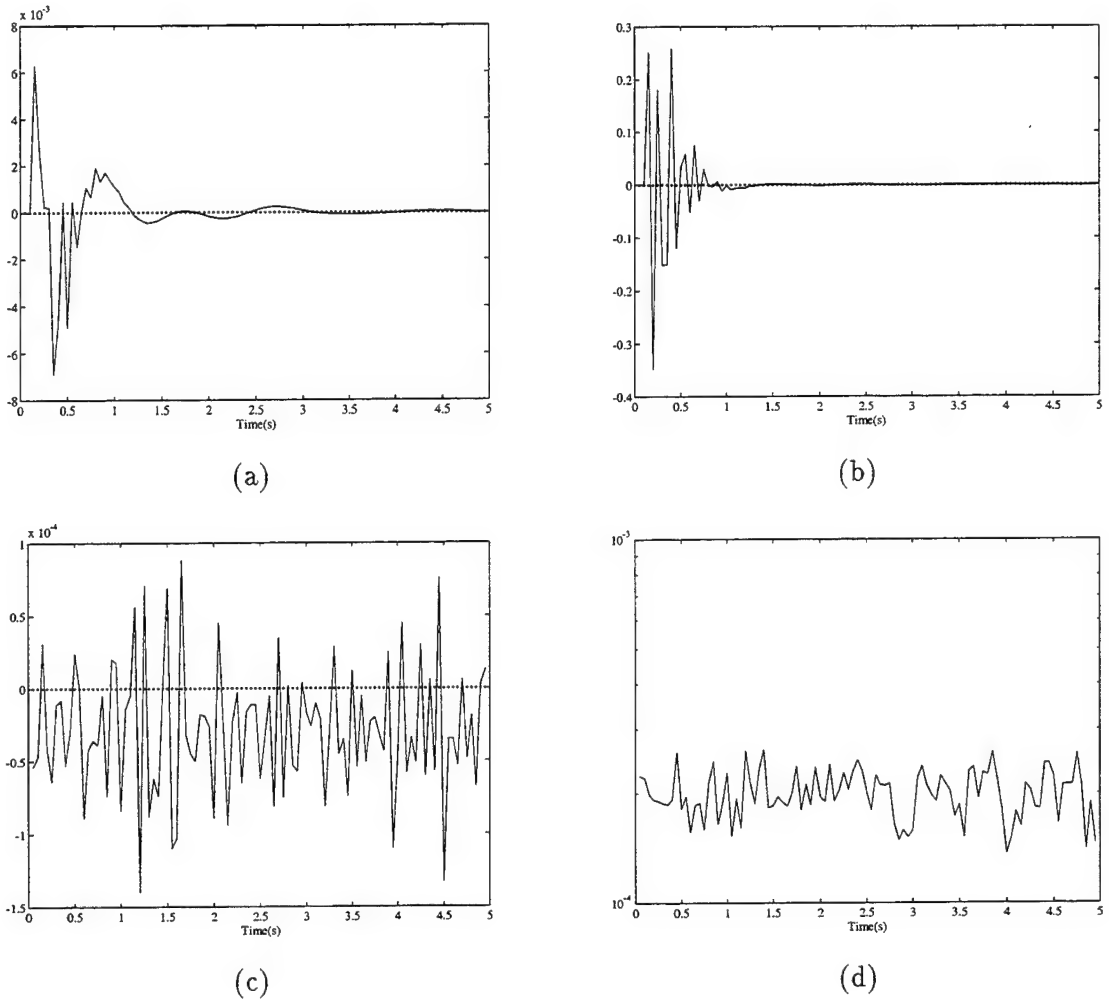


Figure 4: Kinetic versus kinematic: (a) $\theta(t)$, (b) $\dot{\theta}(t)$, (c) Bias in the estimates of $\theta_d(t)$, (d) RMSE in the estimates of $\theta_d(t)$.

motion. The resulting bias and Root-Mean-Squared Error (RMSE) in the estimates of the attitude change between t and $t + \Delta t$, $\theta_d(t) \equiv \theta(t) - \theta(t - \Delta t)$, are shown in Figures 4c and 4d. As seen from the figures, when reliable image cues are available, the performance of the kinematic-law based estimator (dashed line) is very close to that of the kinetic-law based one (solid line). More importantly, the larger bias in the kinematic-law based estimator, due to the modeling error in the time evolution of the parameters, is negligible.

Table 2: Distant point locations.

Point	3-D coordinates		
1	84	-84	1200
2	160	-80	2000
3	170	115	2300
4	100	140	1750

6.3 Real Imagery

We show the results of the application of the stabilization schemes to two sequences with significant unstabilized components in both calibrated and uncalibrated cases. To let the reader more precisely judge the results yielded by this technique, we have placed MPEG movies showing the original sequences, the primitive tracking and the stabilized sequences at the following URL:

http://www.cfar.umd.edu/~yao/stabilization.html.

They can easily be viewed using any web-browser (Mosaic, NetScape, etc).

6.3.1 Martin Marietta Sequence

The first sequence, distributed by Martin Marietta, was obtained from a vehicle performing off-road navigation. Figure 5 shows four frames from the sequence in which each frame has size 347×238. The motion is composed of translation (with a dominant looming component) and unstabilized rotation. The FOV is 40×30 degrees (in the horizontal and vertical directions, respectively), while the optical axis intersects the image plane at the center of the image. The experiments were first carried out for the calibrated case. In this sequence, according to the detection scheme, both the mountain profile and the line near the horizon are detected. Figure 6a shows the results for one frame. (The mountain profile is not employed by the current algorithm.) Four feature points close to the detected line are identified and tracked over the sequence afterwards. For display purposes, the resulting image plane trajectories are superimposed on the last frame of the sequence as shown in Figure 6b. The LS estimate of the rotation is also computed as if only the horizon line were available. Subsequently, stabilization from both full and LS rotation is applied; the recursive algorithm, in addition, uses the estimates from the batch algorithm computed from the first ten frames. Figures 6c and 6d respectively show the LS and full angular velocity estimates. The solid line, the dashed line and the dashed-dotted line respectively correspond to the pitch motion, the

yaw motion and the roll motion. Since the vehicle exhibits very small lateral motion and the sequence is densely sampled, the LS estimates are therefore close to the full estimates for this sequence. Finally, a stabilized sequence is obtained using the global stabilization scheme with the full rotational estimates. The stabilized sequence appears to undergo translation only, with the direction of translation varying noticeably when the rotational component in the original sequence is large.

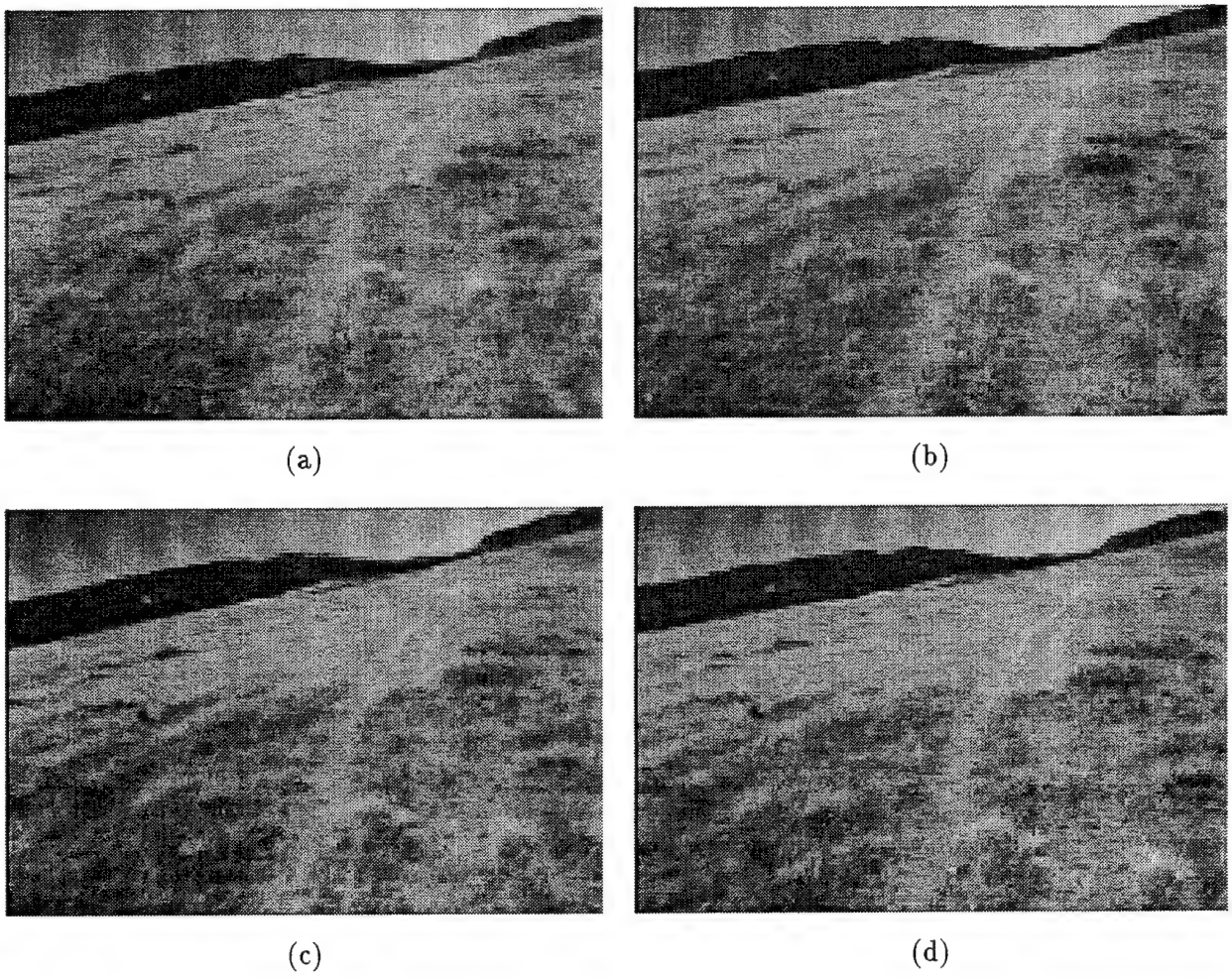


Figure 5: Martin Marietta sequence: (a) Image 9, (b) Image 39, (c) Image 74, (d) Image 99.

The stabilization scheme for the uncalibrated case was also applied, assuming that the intrinsic parameters are unknown. The sensitivity of the eight parameters with respect to the four intrinsic parameters, obtained using the calibrated parameters as nominal values, are first shown in Figures 7

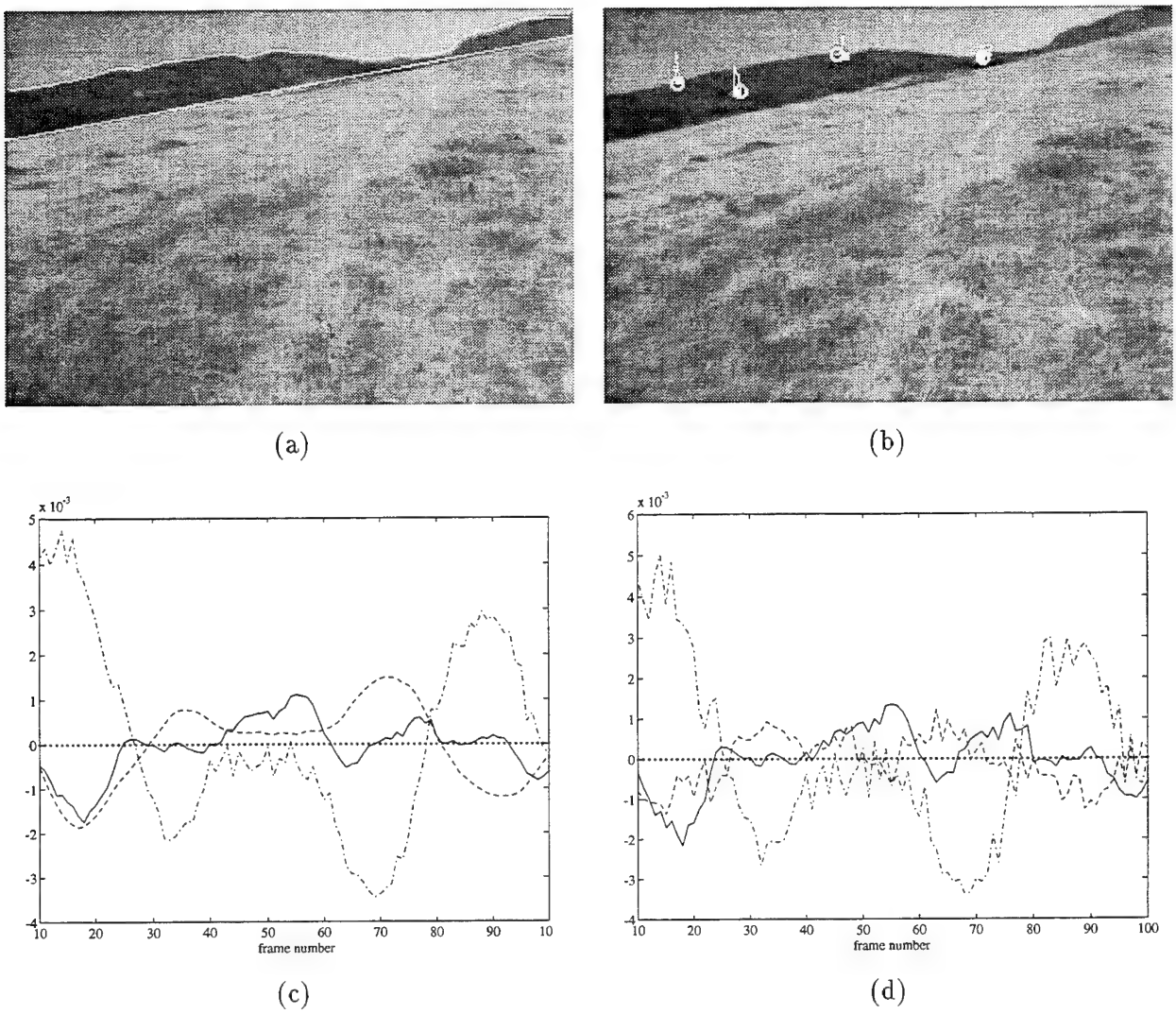


Figure 6: Experimental results from the calibrated scheme for the Martin Marietta sequence: (a) The detected line and the mountain profile, (b) Feature point trajectories, (c) The LS estimates: ω_x (solid line), ω_y (dashed line), ω_z (dashed-dotted line), (d) The full-stabilization estimates.

and 8.[6] (The additional parameter is due to the different focal length in either direction.) For visual purposes, we only illustrate the sensitivity of each parameter with respect to the focal length in the horizontal direction and x_c. As argued earlier, these values are small and therefore a reasonable deviation from the nominal values of the intrinsic parameters is not critical. Consequently, we vary the parameters and perform different tests. For example, assume the focal length in pixels to be 700 and 600 in the horizontal and vertical directions respectively (the real values being 480 and

[6]Instead of showing the computed $\mathbf{J}$, Figures 7 and 8 show the *relative sensitivity*, defined as $\mathbf{J} \cdot \mathbf{\Lambda}$, with $\mathbf{\Lambda}$ a diagonal matrix whose elements are λ_0.

443 respectively), and the image center to be 50 pixels away from the true position, say (100, 100). Then using the same visual cues, Figure 9 shows the estimated angular velocity. The uncalibrated stabilization scheme is thereafter applied to the original sequence. It can be seen from the MPEG movie that the result is close to the one obtained using the calibrated scheme. The result for the uncalibrated sequence using unconstrained projectivity parameters, i.e. direct estimation of the eight parameters from the image features, leads instead very poor results, since the estimation is unstable.

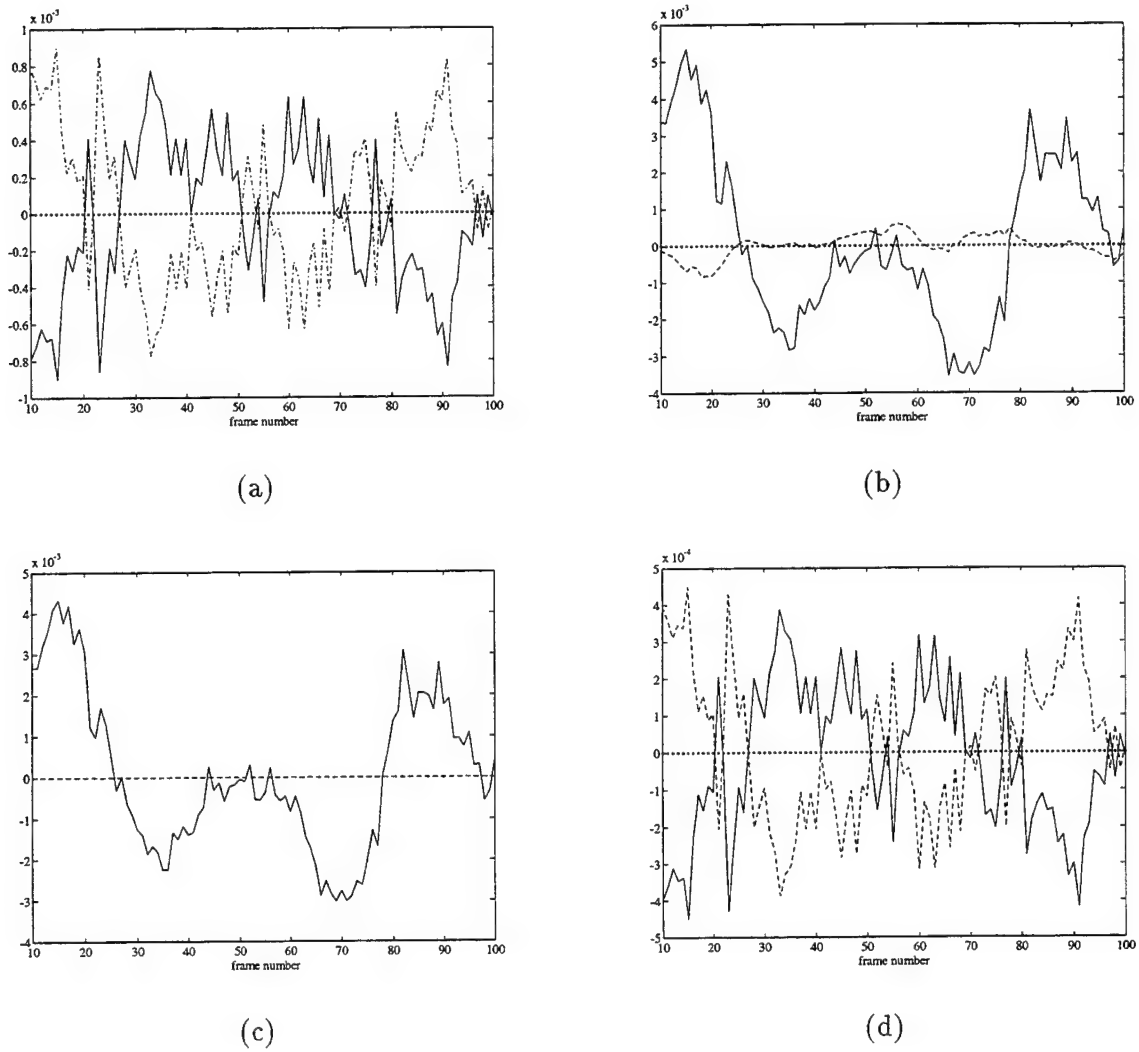


Figure 7: Sensitivities of **A** with respect to f_c in the horizontal direction (solid line) and x_c (dashed line) for the Martin Marietta sequence: (a) a_{11}, (b) a_{12}, (c) a_{21}, (d) a_{22}.

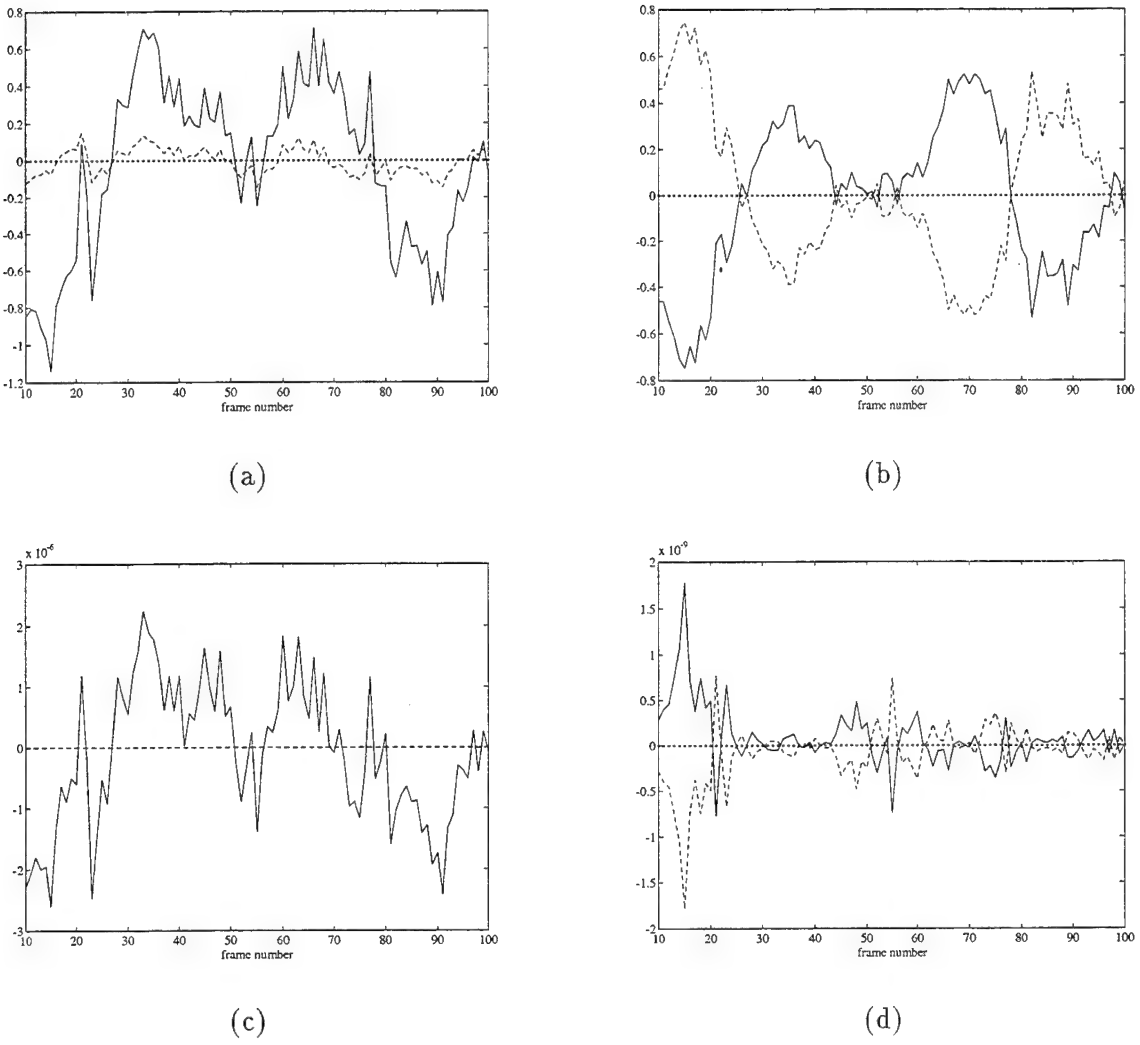
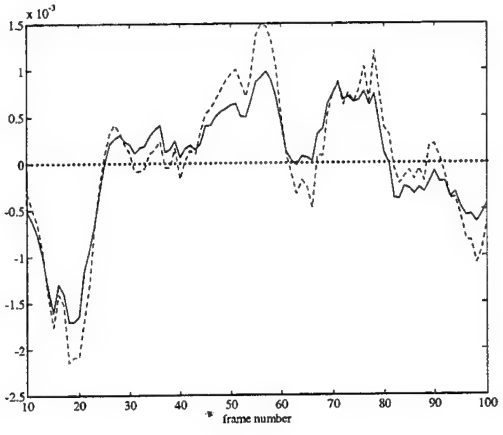


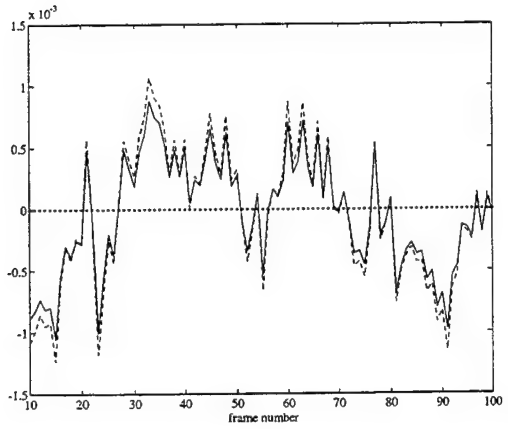
Figure 8: Sensitivities of **b** and **c** with respect to f_c in the horizontal direction (solid line) and x_c (dashed line) for the Martin Marietta sequence: (a) b_1, (b) b_2, (c) c_1, (d) c_2.

6.3.2 NIST Sequence

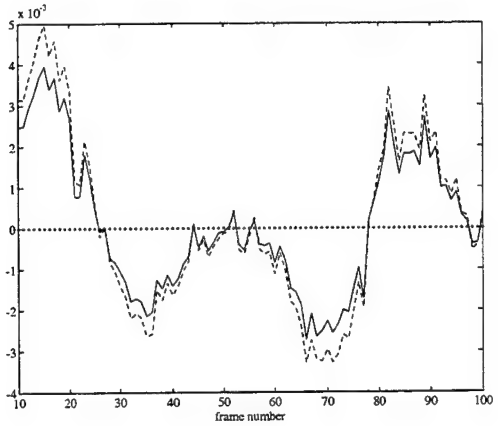
The same procedure was applied to another sequence, provided by NIST, which was also acquired from an off-road vehicle. Figure 10 shows four frames of the sequence in which each frame is of size 640 $\times$ 480. The motion is composed of a translation with steering and unstabilized motion components. The FOV is 70 and 60 degrees along the horizontal and vertical directions respectively, while the optical axis again intersects the image plane at the center of each image. One line and partial object profiles near the horizon are detected as displayed in Figure 11a, while the feature trajectories are similarly superimposed on the last frame as shown in Figure 11b. The LS



(a)



(b)



(c)

Figure 9: Angular velocity estimates from the uncalibrated scheme (solid line) versus calibrated scheme (dashed line) for the Martin Marietta sequence: (a) ω_x, (b) ω_y, (c) ω_z.

and full estimates of the angular velocity are plotted in a similar fashion in Figures 11c and 11d respectively. Since the sequence exhibits some steering, the LS and full estimates are quite different. In addition, the steering behavior is preserved in the stabilized sequence; only the unstabilized motion is removed.

We also applied the uncalibrated stabilization scheme to this sequence. Again, we choose the focal length in pixels to be 600 and 700 (in contrast with the true values 249 and 457), and the center of the image to be (250, 300). The sensitivity of the projective coefficients is shown in Figures 12 and 13. The angular velocity estimates are illustrated in Figure 14. The MPEG movie

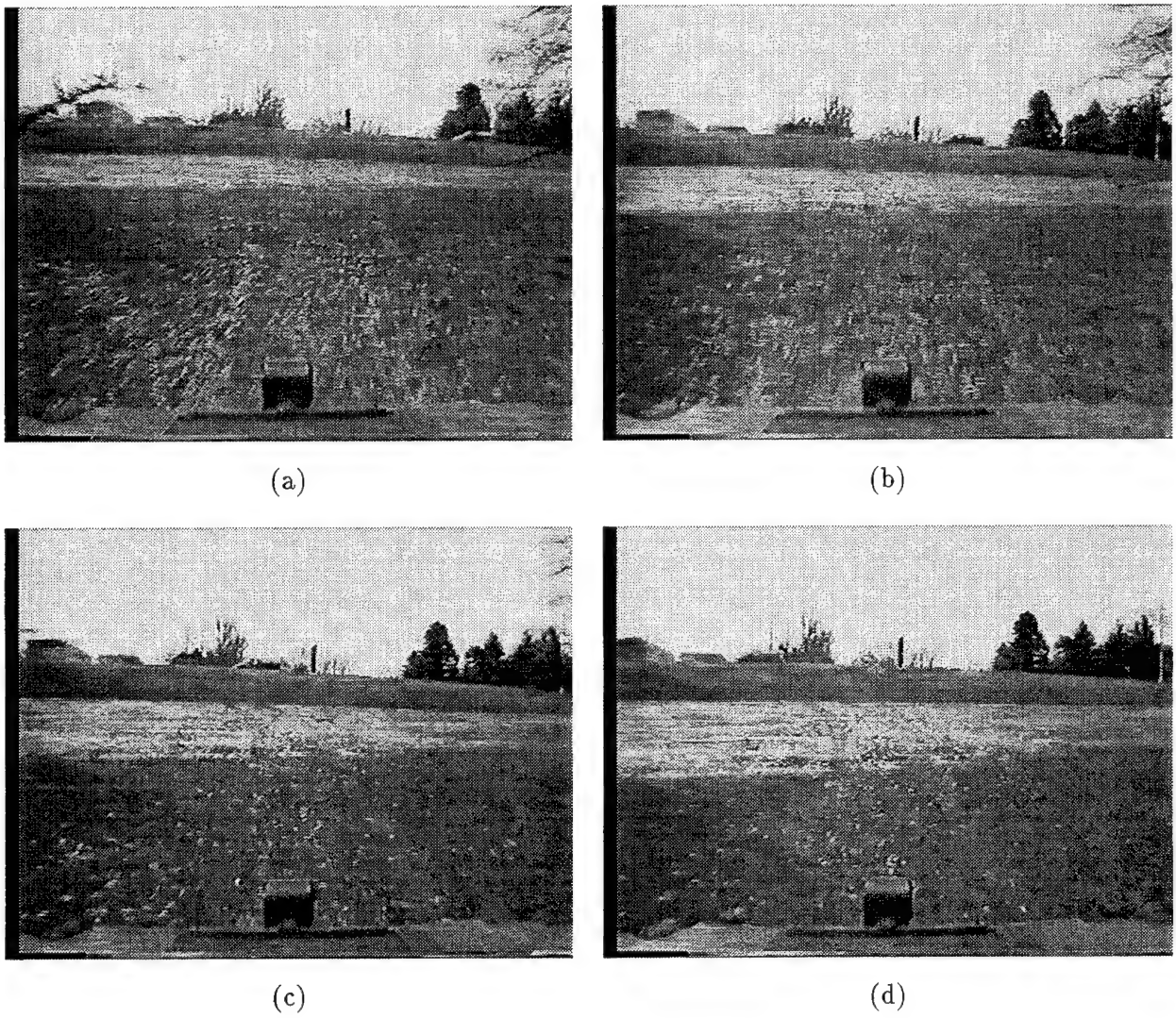


Figure 10: NIST sequence: (a) Image 10, (b) Image 35, (c) Image 60, (d) Image 80.

again shows good stabilization results for this sequence.

7 Conclusion

This paper has presented a scheme for stabilizing a sequence acquired by a moving observer. In particular, the algorithm exploits the temporal information in the sequence and utilizes various visual cues. Image warpings derived from the 3-D motion are used instead of other approximate 2-D mappings, thereby capturing more closely the image motion resulting from the camera rotation. The consideration of global stabilization, in addition, makes the algorithm more suitable to real applications. We analyze the nature of the resulting sequence. The use of kinetic laws for estimation

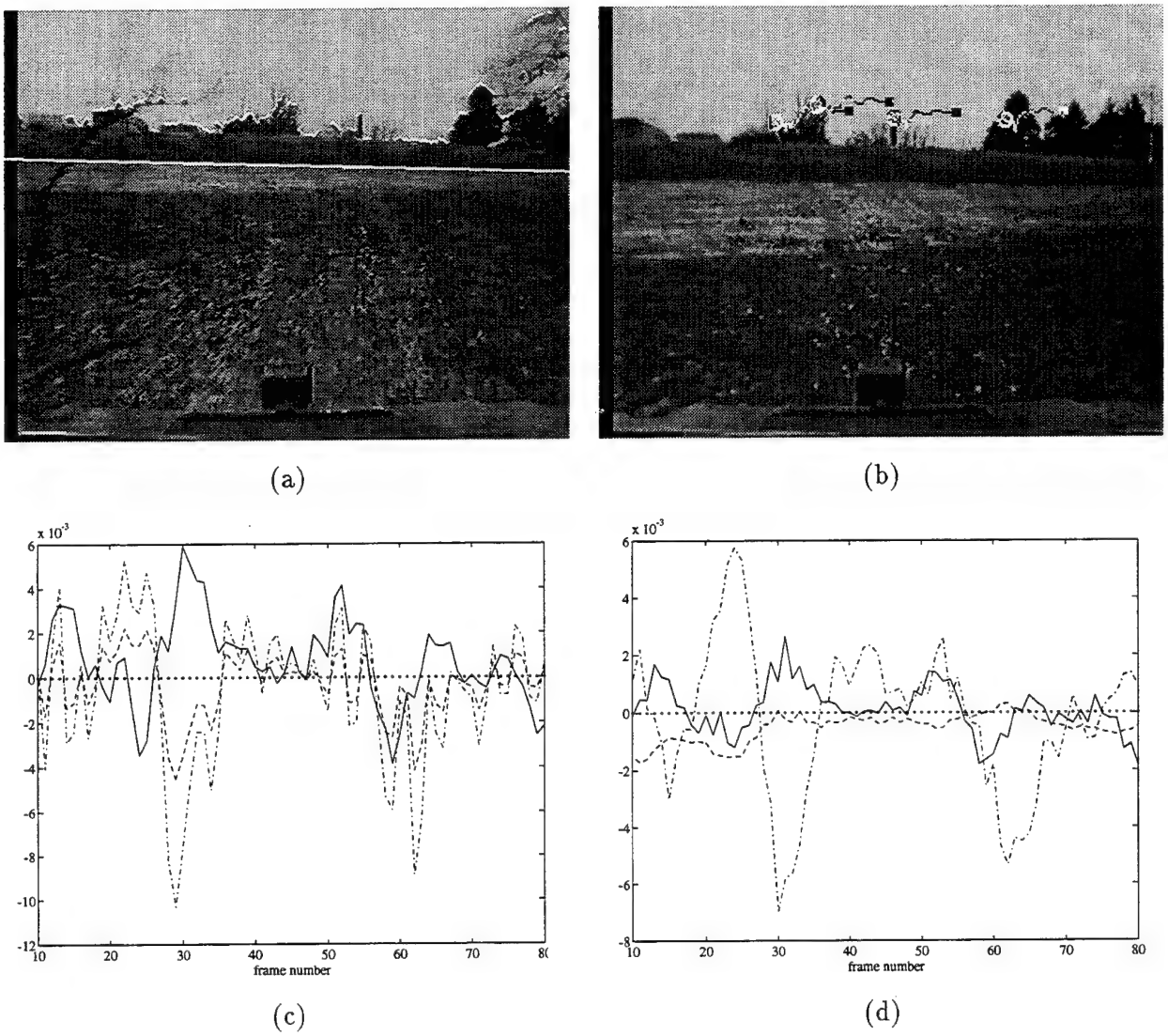


Figure 11: Experimental results from the calibrated scheme for the NIST sequence: (a) The detected line and partial object profiles, (b) Feature point trajectories, (c) The LS estimates: ω_x (solid line), ω_y (dashed line), ω_z (dashed-dotted line), (d) The full-stabilization estimates.

purposes is carefully justified and tested in synthetic cases. The introduction of kinetic laws leads us to study the feasibility of selective stabilization schemes, or the exclusive removal of oscillatory motion components. Lastly, we study the removal of rotational motion from uncalibrated sequences, for which a scheme to alleviate the unstable estimation of the projectivity coefficients is proposed and tested. All results have been thoroughly tested and the resulting sequences have been made available to the interested reader.

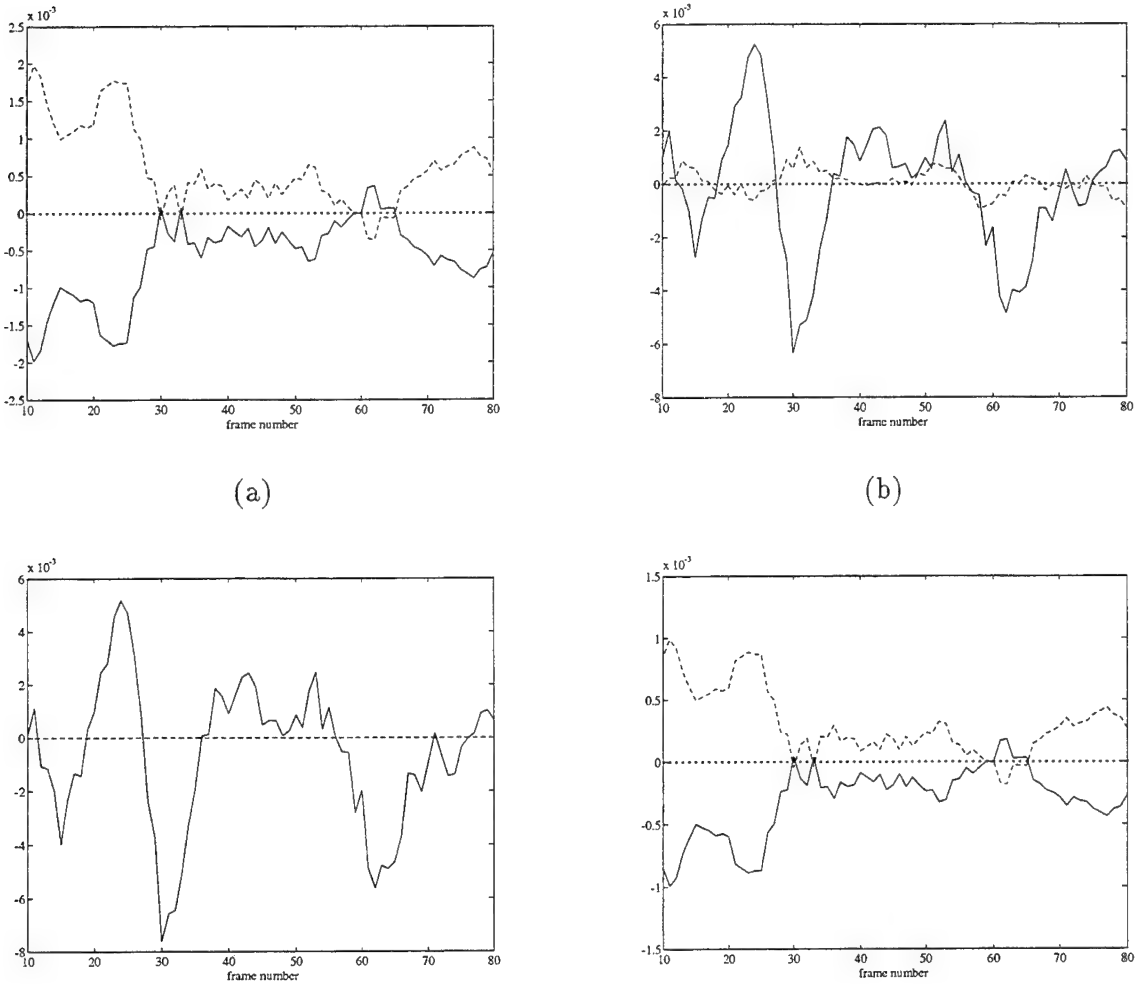


Figure 12: Sensitivities of **A** with respect to f_c in the horizontal direction (solid line) and x_c (dashed line) for the NIST sequence: (a) a_{11}, (b) a_{12}, (c) a_{21}, (d) a_{22}.

References

[1] G. Adiv. Determining three-dimensional motion and structure from optical flow generated by several moving objects. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 7:384–401, July 1985.

[2] P. Burt and P. Anandan. Image stabilization by registration to a reference mosaic. In *Proc. of ARPA Image Understanding Workshop*, pages 425–434, Monterey, CA, November 1994.

[3] P. Burt, R. Hingorani, and R.J. Kolczynski. Mechanisms for isolating component patterns in

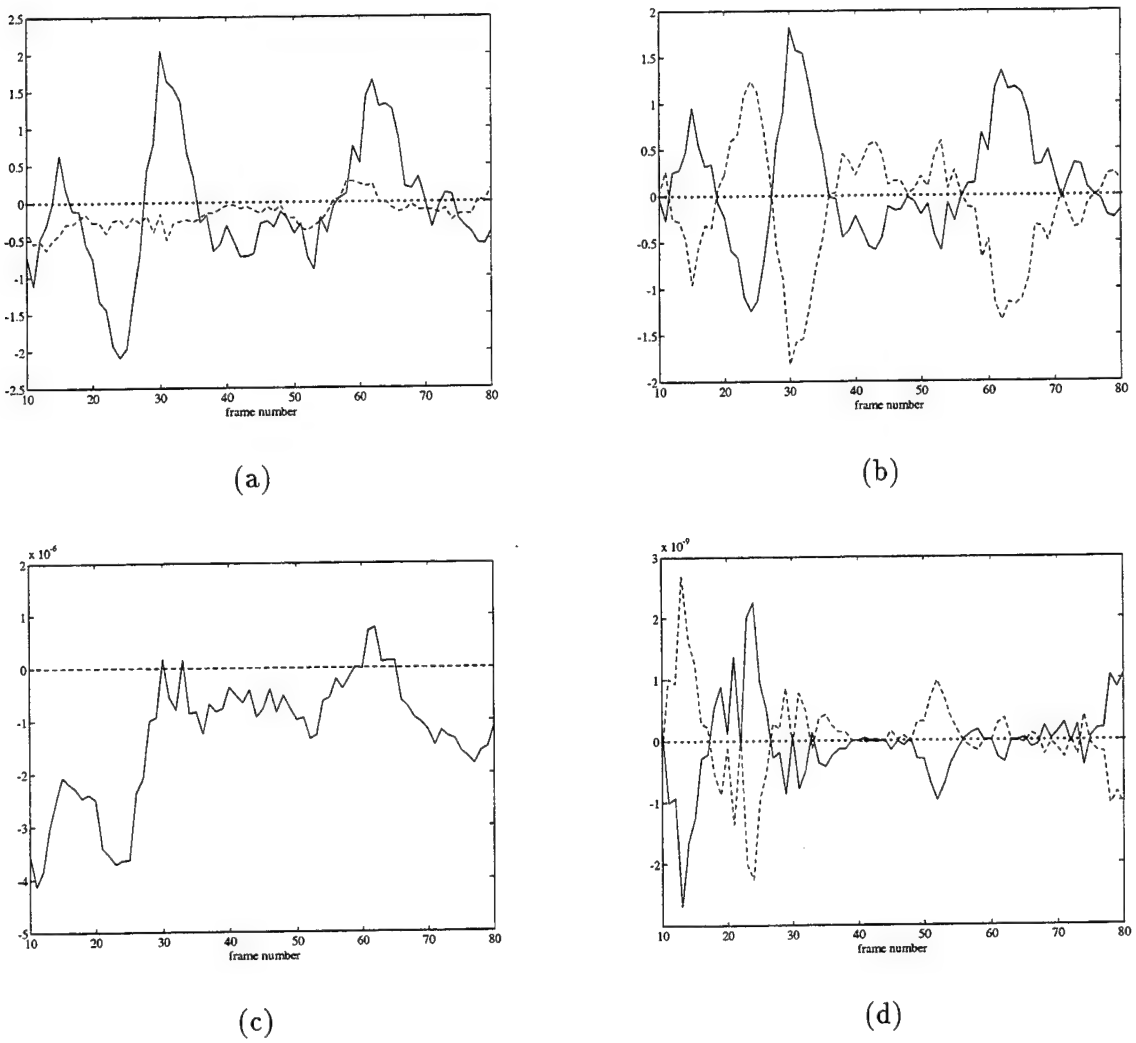


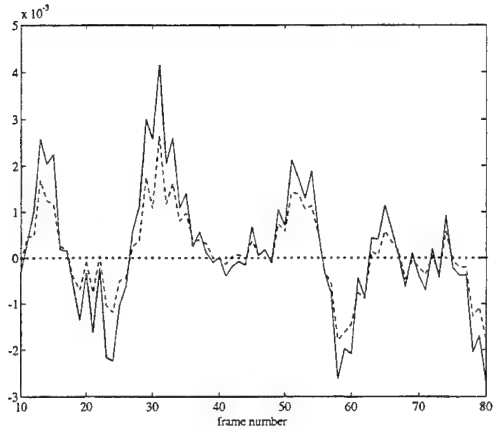
Figure 13: Sensitivities of **b** and **c** with respect to f_c in the horizontal direction (solid line) and x_c (dashed line) for the NIST sequence: (a) b_1, (b) b_2, (c) c_1, (d) c_2.

the sequential analysis of multiple motion. In *Proc. of IEEE Workshop on Visual Motion*, pages 187–193, Princeton, NJ, October 1991.

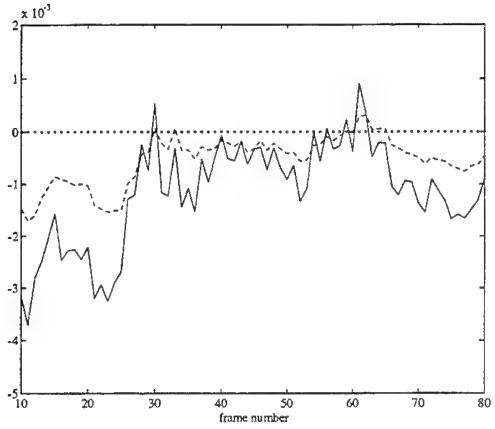
[4] Y.L. Chang and J.K. Aggarwal. 3d structure reconstruction from an ego motion sequence using statistical estimation and detection theory. In *Proc. of IEEE Workshop on Visual Motion*, pages 268–273, Princeton, NJ, October 1991.

[5] C.T. Chen. *Linear System Theory and Design.* Holt, Rinehart, and Winston, New York, 1984.

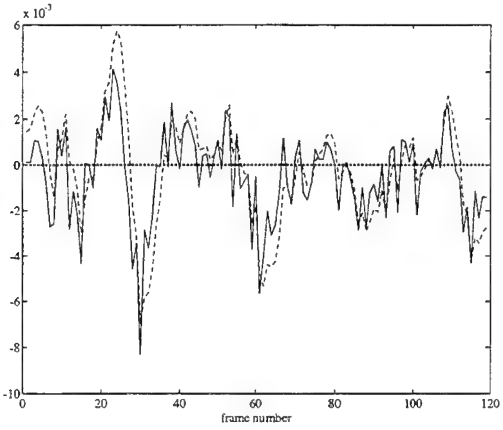
[6] L.S. Davis, R. Bajcsy, R. Nelson, and M. Herman. Rsta on the move. In *Proc. of ARPA Image*



(a)



(b)



(c)

Figure 14: Angular velocity estimates from the uncalibrated scheme (solid line) versus calibrated scheme (dashed line) for the NIST sequence: (a) ω_x, (b) ω_y, (c) ω_z.

Understanding Workshop, pages 435–456, Monterey, CA, November 1994.

[7] A. Hady and D.A. Crolla. Theoretical analysis of active suspension performance using a four-wheel vehicle model. *Proc. of the Institution of Mechanical Engineers*, 203:125–135, 1989.

[8] M. Hansen, P. Anandan, K. Dana, G. van der Wal, and P. Burt. Real-time scene stabilization and mosaic construction. In *Proc. of ARPA Image Understanding Workshop*, pages 457–465, Monterey, CA, November 1994.

[9] M. Irani, B. Rousso, and S. Peleg. Recovery of ego-motion using image stabilization. In *Proc.*

of IEEE Conf. on Computer Vision and Pattern Recognition, pages 454–460, Seattle, WA, June 1994.

[10] R. Kumar, P. Anandan, and K. Hanna. Shape recovery from multiple views: A parallax based approach. In *Proc. of ARPA Image Understanding Workshop*, pages 947–956, Monterey, CA, November 1994.

[11] O.J. Kwon, R. Chellappa, and C. Morimoto. Motion compensated subband coding of video acquired from a moving platform. In *Proc. of IEEE International Conf. on Acoustics, Speech, and Signal Processing*, Detroit,MI, 1995.

[12] R. Nelson. Qualitative detection of motion by a moving observer. *International Journal of Computer Vision*, 7:33–46, 1991.

[13] M. Subbarao. Bounds on time-to-collision and rotational component from first order derivatives of image flow. *Computer Vision, Graphics and Image Processing*, 50:329–341, 1990.

[14] K. Uomori, A. Morimura, H. Ishii, T. Sakaguchi, and Y. Kitamura. Automatic image stabilization system by full-digital signal processing. *IEEE Trans. on Consumer Electronics*, 36:510–519, August 1990.

[15] T. Vieville, E. Clergue, and P.E.D.S. Facao. Computation of ego-motion and structure from visual and internal sensors using the vertical cue. In *Proc. Fourth International Conference on Computer Vision*, pages 591–598, Berlin, Germany, 1993.

[16] Y.S. Yao and R. Chellappa. Model-based vehicular motion and structure estimation. In *Proc. of International Conference on Robotics and Automation*, Nagoya, Japan, 1995.

[17] Y.S. Yao and R. Chellappa. Tracking a dynamic set of feature points. *IEEE Trans. on Image Processing*, October 1995.

[18] Z. Zhang and O.D. Faugeras. Three-dimensional motion computation and object segmentation in a long sequence of stereo frames. *International Journal of Computer Vision*, 7:211–241, August 1992.

[19] Q. Zheng and R. Chellappa. Automatic feature point extraction and tracking in image sequences for arbitrary camera motion. *International Journal of Computer Vision*, 1995.

[20] Y.J. Zheng, W. Ritter, and R. Jansen. An adaptive system for traffic sign recognition. In *Proc. of Intelligent Vehicles Symposium*, pages 165–170, Paris, France, 1994.

Appendix A

We give the derivation of image motion in vector form in this appendix. As in Section 2, assume that perspective projection is used as the imaging model and the camera undergoes translation and rotation. Then, for a 3-D point $\mathbf{P} : (X, Y, Z)^T$, the image motion of the projection point $\mathbf{p} : (x, y)^T$ can be obtained as follows:

$$\dot{\mathbf{p}} = \frac{\partial \mathcal{P}}{\partial \mathbf{P}} \dot{\mathbf{P}} + \dot{\mathbf{p}}_c \tag{75}$$

$$= f_c \begin{bmatrix} \frac{1}{Z} & 0 & -\frac{X}{Z^2} \\ 0 & \frac{1}{Z} & -\frac{Y}{Z^2} \end{bmatrix} \dot{\mathbf{P}} \tag{76}$$

$$= \frac{1}{Z} \begin{bmatrix} f_c & 0 & -(x - x_c) \\ 0 & f_c & -(y - y_c) \end{bmatrix} (-\boldsymbol{\omega} \times \mathbf{P} - \mathbf{V}) \tag{77}$$

$$= f_c^{-1} \begin{bmatrix} f_c & 0 & -(x - x_c) \\ 0 & f_c & -(y - y_c) \end{bmatrix} \left\{ \begin{bmatrix} 0 & \omega_z & -\omega_y \\ -\omega_z & 0 & \omega_x \\ \omega_y & -\omega_x & 0 \end{bmatrix} \begin{bmatrix} x - x_c \\ y - y_c \\ f_c \end{bmatrix} - \begin{bmatrix} f_c \frac{V_x}{V_z} \\ f_c \frac{V_y}{V_z} \\ f_c \end{bmatrix} \frac{V_z}{Z} \right\} \tag{78}$$

As before, define $\mathbf{p}_{\text{foe}} : (f_c \frac{V_x}{V_z}, f_c \frac{V_y}{V_z})^T + \mathbf{p}_c$ and $\tau : \frac{Z}{V_z}$ as well as the following matrices

$$\boldsymbol{\Omega}_z = \begin{bmatrix} 0 & \omega_z \\ -\omega_z & 0 \end{bmatrix} \quad \boldsymbol{\omega}_{xy}^{\perp} = \begin{bmatrix} -\omega_y \\ \omega_x \end{bmatrix} \tag{79}$$

Then (78) can be rewritten as

$$\dot{\mathbf{p}} = f_c^{-1} \left[\begin{array}{c|cc} f_c \mathbf{I} & -(\mathbf{p} - \mathbf{p}_c) \end{array} \right] \left\{ \left[\begin{array}{cc|c} \boldsymbol{\Omega}_z & & \boldsymbol{\omega}_{xy}^{\perp} \\ - & - & - \\ -\boldsymbol{\omega}_{xy}^{\perp T} & & 0 \end{array} \right] \begin{bmatrix} \mathbf{p} - \mathbf{p}_c \\ - \\ f_c \end{bmatrix} - \begin{bmatrix} \mathbf{p}_{\text{foe}} - \mathbf{p}_c \\ - \\ f_c \end{bmatrix} \frac{1}{\tau} \right\} \tag{80}$$

Consequently, we have

$$\dot{\mathbf{p}} = f_c^{-1} (\mathbf{p} - \mathbf{p}_c) \boldsymbol{\omega}_{xy}^{\perp T} (\mathbf{p} - \mathbf{p}_c) + \boldsymbol{\Omega}_z (\mathbf{p} - \mathbf{p}_c) + f_c \boldsymbol{\omega}_{xy}^{\perp} - (\mathbf{p}_{\text{foe}} - \mathbf{p}) \frac{1}{\tau} \tag{81}$$

Appendix B

We derive the relationship between $\mathbf{w}$, the projection of the 3-D normal vector to the plane Π (see Figure 2), and the image plane normal $\mathbf{o}$ in this appendix. Consider an image horizon line denoted by

$$\mathcal{L} : y = a + bx \tag{82}$$

where $(x, y)^T$ are the image plane coordinates of a 3-D point. Then the image plane vector normal to $\mathcal{L}$, defined in (20), is

$$\mathbf{o} = (-\frac{b}{a}, \frac{1}{a})^T \tag{83}$$

Let f_c be the focal length of the camera and assume that the optical axis intersects the image plane at $\mathbf{p}_c : (x_c, y_c)^T$. Then

$$\begin{array}{rcl} \mathbf{P}_1 & : & (0, a + bx_c - y_c, f_c)^T \\ \mathbf{P}_2 & : & (\frac{y_c - a - bx_c}{b}, 0, f_c)^T \end{array} \tag{84}$$

are the coordinates of two points lying on the plane Π, with respect to the camera frame of reference. Consequently, by taking the cross product of $\mathbf{P}_1$ and $\mathbf{P}_2$, we have

$$\mathbf{N} \stackrel{\text{def}}{=} \mathbf{P}_1 \times \mathbf{P}_2 \tag{85}$$

$$= (1, -\frac{1}{b}, \frac{a + bx_c - y_c}{bf_c})^T \tag{86}$$

The 3-D normal vector $\mathbf{W}$ is then obtained by

$$\mathbf{W} = \frac{\mathbf{N}}{||\mathbf{N}||} \tag{87}$$

$\mathbf{w}$ is related to $\mathbf{o}$ by

$$\mathbf{w} = \mathcal{P}(\mathbf{W}) + \mathbf{p}_c \tag{88}$$

$$= \begin{bmatrix} f_c^2 \frac{b}{a + bx_c - y_c} \\ -f_c^2 \frac{1}{a + bx_c - y_c} \end{bmatrix} + \mathbf{p}_c \tag{89}$$

$$= -\frac{f_c^2}{1 - < \mathbf{p}_c, \mathbf{o} >} \mathbf{o} + \mathbf{p}_c \tag{90}$$

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	July 1995	Technical Report

4. TITLE AND SUBTITLE

3-D Model-Based Image Stabilization Using Multiple Visual Cues

5. FUNDING NUMBERS

DAAH04-93-G-0419

6. AUTHOR(S)

Y.S. Yao, P. Burlina, R. Chellappa, and T.H. Wu

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Computer Vision Laboratory
Center for Automation Research
University of Maryland
College Park, MD 20742-3275

8. PERFORMING ORGANIZATION REPORT NUMBER

CAR-TR-781
CS-TR-3506

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This paper studies the problem of image stabilization, defined here as the process of generating a compensated video sequence where image motion resulting from camera motion has been partially or totally removed. The scheme combines various visual cues such as points and horizon lines, and relies on an Extended Kalman Filter for the estimation of parameters of interest. We study both calibrated and uncalibrated stabilization cases. We address the issues of local versus global stabilization. We consider the problem of the selection of model dynamics for the estimation of warping parameters and illustrate the use of kinetic models for the selective removal of oscillatory motion. Experimental results from video sequences generated from off-road vehicle platforms show good performance of the stabilization schemes.

14. SUBJECT TERMS

Image stabilization, motion analysis, image warping, integration of visual cues, kinematic and kinetic laws

15. NUMBER OF PAGES

42

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL